AD-A103 285 UNITED TECHNOLOGIES CORP WEST PALM BEACH FLA F/G 20/5
SYSTEM OPTICAL QUALITY USERS GUIDE. PART 2.(U)
MAR 80 J L FORGHAM, S S TOWNSEND F29601-77-C-0025
UNCLASSIFIED AFWL-TR-79-141-PT-2 NL

1 of 3
AD A
103285

AD-E200 723

AFWL-TR-79-141 Pt-2

(2) LEVEL III

AFWL-TR-
79-141
Pt. 2

SYSTEM OPTICAL QUALITY USERS GUIDE

Part 2,[redacted]

J.L. Forgham S. S. Townsend J. L. Campbell

United Technologies Corporation
West Palm Beach, FL 33402

Mar[redacted]80

Final Rep[redacted]t

Approved for public release; distribution unlimited.

AIR FORCE WEAPONS LABORATORY
Air Force Systems Command
Kirtland Air Force Base, NM 87117

This final report was prepared by the United Technologies Corporation, West Palm Beach, Florida, under Contract F29601-77-C-0025, Job Order 00011408 with the Air Force Weapons Laboratory, Kirtland Air Force Base, New Mexico. Captain J. Dale Holt (ARLO) was the Laboratory Project Officer.

This technical report has been reviewed and is approved for publication.

J. Dale Holt

J. DALE HOLT
Captain, USAF
Project Officer

FOR THE DIRECTOR

James L. Stapp

JAMES L. STAPP
LtCol, USAF
Chief, Optical Systems Branch

John A. Carpenter

JOHN A. CARPENTER
Colonel, USAF
Deputy Chief, Laser Development Division

DISCLAIMER NOTICE

SOQ USER GUIDE UPDATES

June 1980 Updates to SOQ80128

INTRODUCTION

This document defines the changes made to the SOQ code (SOQ80128) between January and June of 1980. The changes either correct shortcomings found in the code or, more usually, document the increased capability being continually built into the code. The SOQ code is maintained as SOQ80128 June PL,ID = AFLOJRA as a NOS/BE-1 CDC update format file.

UPDATES

1. *ID FIXZRN

This update redefines the coeffiencets to be input to the Zernike subroutine. This new convention is more physically meaningful in that, at least for lower orders, the coefficients are in waves. For example, to impose one wave peak to peak of defocus (P_4) on a beam, one would input P(4)=1. The phase applied is now:

$$\phi(I,J) = \sum_k P_k \pi Z_k(I,J)$$

The subroutine affected is ZERN. This update does not effect the rest of the code.

2. *ID FIXJTR

This update ensures a correct definition of DF in subroutine JITRBG since when JITRBG is called from subroutine QUAL, the X-coordinate array contains $R\lambda/D$ coordinates, not the spatial coordinates.

Only one line of the code is affected by this update.

3. *ID ROTZRN

Due to different coordinate system orientations for data, it became necessary to allow for this variation within subroutine ZERN.

Define the data x and y coordinates to be XROT and YROT, and the SOQ x and y coordinates to be XIN and YIN. The rotation angle is then defined to be θ (in radians).

$$COSROT = COS(\theta)$$

$$SINROT = SIN(\theta)$$

$$XROT = XIN \times COSROT + YIN \times SINROT$$

$$YROT = -XIN \times SINROT + YIN \times COSROT$$

Application of Zernike polynomials to and SOQ point located at (XIN, YIN) would then be calculated using Z(XROT, YROT). The possibility of axis flips are also accounted for and are flagged by FLIPX or FLIPY not equal to zero. Namelist ZERNS is modified to include FLIPX, FLIPY and the rotation angle (in degrees) ZTHETA. No common was modified. This update modified only subroutines GDL and ZERN.

```
    *IDENT FIXZRN

       */  ZERN
         *DELETE ZRNIKE.115
               DEL = DFL*3.14159264
         *DELETE ZRNIKE.125
              C 20X,22H PHI(N) = PI*R(N)*Z(N)//

    *IDENT FIXJTR
       */  JITREG
         *DELETE JITTER.29,JITTER.30
               DF = 1./(FLOAT(NPTS)*DX)

    *IDENT ROTZRN
       */  SOL
         *DELETE ZRNINPO.3
               NAMELIST /ZERNS/ RO,F,FRPNG,SIGMAY,NTERMZ,ZTHETA,FLIPX,FLIPY
         *INSERT ZRNIKE.9
         C           ZTHETA = THE CLOCKWISE ANGLE OF ROTATION OF THE DECOMPOSITIO
         C                    AXES ONTO THE SOG COORDINATE SYSTEM
         C                    BEFORE CALCULATION OF THE ZERNIKE POLYNOMIALS.
         C                    IT IS INPUT IN DEGREES.
         C           FLIPX = 1. RESULTS IN A FLIP ABOUT THE X AXIS BEFORE
         C                    ROTATION.
         C           FLIPY = 1. RESULTS IN A FLIP ABOUT THE Y AXIS BEFORE
         C                    ROTATION.
         *DELETE ZRNINPO.2
               DIMENSION PZZSV(20,10)
         *INSERT ZRNINPO.7
               ZTHETA = 0.
               FLIPX = 0.
               FLIPY = 0.
         *INSERT ZRNINPO.9
               PZZSV(IZERN,3) = ZTHETA*3.141593/180.
               PZZSV(IZERN,4) = FLIPX
               PZZSV(IZERN,5) = FLIPY
         *DELETE ZRNINPO.10,ZRNINPO.11
           244 CALL ZERN(PZZSV(IZERN,1),PZZSV(IZERN,2),PZZSV(IZERN,3),
              X           PZZSV(IZERN,4),PZZSV(IZERN,5),
              X           PZSAVE(25,IZERN),PZSAVE(1,IZERN))
       */  ZERN
         *DELETE ZRNINPO.12
               SUBROUTINE ZERN(SIGMAY,XNTERMZ,THETA,FLIPX,FLIPY,RO,F)
         *INSERT ZRNIKE.72
               COSROT = COS(THETA)
               SINROT = SIN(THETA)
         *DELETE ZRNIKE.75
         *DELETE ZRNIKE.77
               XIN = X(IX)
               YIN = X(IY)
               IF(FLIPX.GT..5) YIN=-YIN
               IF(FLIPY.GT..5) XIN=-XIN
               XROT =  XIN*COSROT + YIN*SINROT
               YROT = -XIN*SINROT + YIN*COSROT
               IF(FLIPX.LT.-.5) YROT=-YROT
               IF(FLIPY.LT.-.5) XROT=-XROT
               XSQ = XROT**2
               YSQ = YROT**2
         *DELETE ZRNIKE.80
               THET = ATAN2(YROT,XROT)
```

```
*IDENT MCRSLM
      *INSERT SUMMRY.515
      C
      C   **** COPY TAPE(50) TO OUTPUT:
      C
            END FILE 50
      C
            WRITE(6,3035)
            REWIND 50
       7000 READ(50,4005) IC1,C2
       4005 FORMAT(I1,21A4)
            IF(EOF(50).NE.0.) GO TO 7015
      C     IF(IC1.EQ.1) WRITE(6,3035)
            WRITE(6,4040) C2
       4040 FORMAT(10X,21A4)
            GO TO 7000
       7015 REWIND 50
            WRITE(6,3035)
      C
            REWIND 97
       4000 READ(97,4005) IC1,C2
            IF(EOF(97).NE.0.) GO TO 4015
            IF(IC1.EQ.1) WRITE(6,3035)
            WRITE(6,4040) C2
            GO TO 4000
       4015 REWIND 97
            WRITE(6,3035)
      C
            REWIND 97
       6000 READ(97,4005) IC1,C2
            IF(EOF(97).NE.0.) GO TO 6015
            IF(IC1.EQ.1) WRITE(6,3035)
            WRITE(6,4040) C2
            GO TO 6000
       6015 REWIND 97
            WRITE(6,3035)
      C
      C   **** COPY TAPE(ISUMRY) TO OUTPUT:
      C
            REWIND ISUMRY
       5000 READ(ISUMRY,3005) IC1,C2,C3
            IF(EOF(ISUMRY).NE.0.) GO TO 5015
            IF(IC1.EQ.1) WRITE(6,3035)
            WRITE(6,3040) C2,C3
            GO TO 5000
       5015 REWIND ISUMRY
            WRITE(6,3035)
      C
      C   **** COPY TAPE(50) TO OUTPUT:
      C
            WRITE(6,3035)
            REWIND 50
       8000 READ(50,4005) IC1,C2
            IF(EOF(50).NE.0.) GO TO 8015
      C     IF(IC1.EQ.1) WRITE(6,3035)
```

```
          WRITE(6,4040) C2
          GO TO 9000
     9015 REWIND 50
          WRITE(6,3035)
    C
```

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE

READ INSTRUCTIONS BEFORE COMPLETING FORM

1. REPORT NUMBER: AFWL-TR-79-141 — 2. GOVT ACCESSION NO.: AD-A103 285	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle): SYSTEM OPTICAL QUALITY USERS GUIDE Part 2 of 3.	5. TYPE OF REPORT & PERIOD COVERED: Final Report 6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s): J. L. Forgham S. S. Townsend J. L. Campbell	8. CONTRACT OR GRANT NUMBER(s): F29601-77-C-0025
9. PERFORMING ORGANIZATION NAME AND ADDRESS: United Technologies Corporation West Palm Beach, FL 33402	10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS: 00011408/63605F
11. CONTROLLING OFFICE NAME AND ADDRESS: Air Force Weapons Laboratory (ARLO) Kirtland Air Force Base, NM 87117	12. REPORT DATE: March 1980 13. NUMBER OF PAGES: 262
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office)	15. SECURITY CLASS. (of this report): Unclassified 15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

This report is divided into three parts. Part 1 consists of the front matter and text pages 1-34. Part 2 consists of text pages 35-296 and the References. Part 3 consists of Appendices A and B and distribution list pages 297-360.

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Laser
Code
Optics
Optical Quality
Optical System
High Power Laser
High Energy Laser

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

This report describes the System Optical Quality (SOQ) code structure and the input to the code required for analyzing High Power Laser Optical Systems. The SOQ code provides the designer with a physical optics model of the system. The code traces the beam from its point of origin in the resonator through the optical train into the far field. This report is divided into three parts. Part 1 describes the general structure of the SOQ code and establishes a correlation between the usual optical elements encountered in the optical — over

DD FORM 1 JAN 73 1473

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

20. ABSTRACT (Continued).

train/gas dynamic laser resonator and the appropriate SOQ models. Part 2 acquaints the user with the individual SOQ subroutines and their analytical formulations as manifested in Fortran within the SOQ framework. It also delineates the input required to exercise the subroutines, familiarizes the user with the operation of the SOQ model, and contains working input modules which carry the user through the usual calculations of the SOQ code from input generation to loaded cavity calculations. Part 3 contains Appendices describing SOQ updates.

SECTION III
MAIN EXECUTIVE CODE

1. PROGRAM SOQ

a. Purpose -- Program SOQ is the main driver program or executive code for the total SOQ code. Many parameters such as mesh size, number of points, initial position of the optical axis, the initial coordinate array, and the initial field itself are established in this routine. Once the above parameters have been initialized, there are several options available for operations on the field. Those available are:

(1) Calling subroutine GDL, the executive program for propagating the field through the optical elements

(2) Performing a quality calculation

(3) Gradient search optimization

(4) Parametric studies.

The above options can be activated in any order and as many times as desired by successive reads of namelist START. The flag for ending execution of the entire deck is to set WWL = 0 in the last read of START.

b. Formalism -- The only major explicit calculations done in SOQ are those which determine the initial field when it is not to be read in. The OPTIONS are:

(1) Plane wave - constant amplitude

(2) Plane wave - Gaussian amplitude

(3) Spherical wave - constant amplitude

(4) Spherical wave - Gaussian amplitude.

Letting $E(x,y)$ represent the field, $A(x,y)$ the field amplitude, and ϕ (x,y) the field phase, then the field is determined by:

$$E(x,y) = A(x,y)\ e^{i\phi(x,y)} \tag{8}$$

where

$$A(x,y) = \begin{cases} \text{const.} \\ \text{const.}\ e^{-\left(\frac{x^2 + y^2}{\sigma^2}\right)} \end{cases} \tag{9}$$

and

$$\phi(x,y) = \begin{cases} 0 \\ e\ -\frac{\pi}{\lambda R}\left(x^2 + y^2\right) \end{cases} \tag{10}$$

The other calculations based on input distributions are performed in subroutines.

c. Fortran -- The only common variables that are not altered in this routine are SPACE and CFIL. The others are altered and are defined as follows:

CU = the complex field array
X = the coordinate array
DRX = the x position of the optical axis
DRY = the y position of the optical axis
NPTS = the number of points in the x direction
NPY = the number of points in the y direction
= NPTS if SYMTRC is false
= NPTS/2 if SYMTRC is true
WL = the wavelength of the radiation
PLTSG = plotting parameter (none, amplitude, or intensity)
INT = set to 0

The relevant parameters are read into the program by means of the namelists described below.

(1) Namelist START -- This namelist is used to initialize parameters such as field, mesh, and coordinates, and is used to direct the calculation to other sections of SOQ. It is read repeatedly until WWL $\leq$ 0 is encountered.

```
C*****************************************************************************C
      NAMELIST /START/ WL,DCAL,NPTS,DORX,DORY,RESTRT,IN,IR,NCALL,
     X AMPGES,  ITNUM,   SYMTRC   ,DGAUSS , TITLES, PHIRAD
     X   , PLOTS
C PLOTS=-1..  AMPLITUDE , PHASE SLICE PLOTS
C PLOTS= 0..  NO SLICE OR ISO-INTENSITY PLOTS
C PLOTS= 1..  INTENSITY, PHASE SLICE PLOTS
C PLOTS=1.. INTENSITY, PHASE SLICE PLOTS          OT(IPLTS)
C       NCALL CONTROLS THE MOVEMENT INSIDE MAIN
C        = 2, GDL SECTION,CALLS GDL AND READ CU,X FROM DISK
C        = 3 CALL TO QUALITY ALGORYTHUM, READS QLOT
C        = 4 CALL TO ANY OF THE GOULD PLOTTING PACKAGES, READS THRED
C        = 5 STARTS OPTIMAZATION ALA DAVION, READS OPTIM
C        = 6 PARAMETRIC STUDIES..INVOLVES CHANGING ARC ARRAY FOR GDL,
C            READS PARAM
C*****************************************************************************C
C       WL IS RADIATION WAVELENGHT
C       DCAL IS INTIAL SIZE OF CALCULATION REGION
C       NPTS IS NUMBER OF FIELD POINTS ACROSS DCAL
C       DRX,DRY = THE (X,Y) POSITION OF THE CENTER OF CU RELATIVE
C                 TO THE OPTICAL AXIS
C       RESTRT IS CONTROL FOR RESTARTING WITH EXISTING GAIN CO-EFF AND
C       INITIAL FIELD FROM PREVIOUS RUN..........
C           .TRUE. IF RESTARTING, OR IF FIELD IS TO BE READ FROM IR
C           .FALSE. IF NOT, OR IF INITIAL FIELD AND X@S ARE TO BE CALC
C       IN = UNIT NUMBER OF DATA SET FOR GDL AND CAVITY
C          IF IN = 0, THEN THERE IS NO CALL TO GDL
C
C       IR IS UNIT NUMBER OF INPUT FIELD TO GDL
C         IF IR = 0, THEN NOTHING IS READ
C       AMPGES IS INITIAL AMPLITUDE OF STARTING BEAM(PEAK AMPLITUDE
C       FOR GAUSSIAN)
C       PHIRAD IS    PHASE FRONT RADIUS OF CURVATURE (=0.0 FOR PLANE)
C       ITNUM IS THE ITERATION NUMBER...IF UNSPECIFIED IT READS OFF DISK
C
C          SYMTRC IS LOGICAL FOR SYMMETRIC ANAYLSYS OR NOT
C       DGAUSS IS DIAMETER AT WHICH GAUSSIAN AMPLITUDE = 1.0/E
C
C*****************************************************************************C
```

(2) Namelist QLOT -- This namelist establishes the parameters necessary to perform quality calculations.

```
       NAMELIST/QLOT/ TITLE,  IQLT, DB, ISAV, IPHASE, R9B ,RF
C
C      TITLE FOR PLOTS IN QUALITY ROUTINE
C      IQLT IS PLOTTING PARAMETER FOR PLTOT....QUALITY PLOTS
C           = 0 ISO-INTENSITY AND POWER VS RL/D GOULD PLOTS
C           = 1 FAR-FIELD PWR VS RL/D GOULD.......
C           = 2 NO GOULD PLOTS  CALCULATES POWER DIST, ONLY
C           = 3 ISO-INTENSITY GOULD, PWR DIST, BUT NO PWR RL/D PLOTS
C           = 4 CALC POWER INSIDE R9B ONLY...NO CALL TO PLTOT
C      DB = BEAM DIAMETER
C      ISAV IS SAVING PARAMETER........
C           =0, DONT SAVE         =1 SAVE INPUT FIELD
C           =-1 USE DATA SET #9 FOR INPUT
```

```
C       IPHASE CONTROLS THE PHASE CORRECTIONS APPLIED TO THE FIELD
C            = 0   NONE
C            = 1   PLANAR CORRECTION
C            = 2   SPHERICAL
C            = 3   BOTH
C          RBB IS THE BUCKET SIZE FOR OPTIMAZATION...IF A CALL TO QUAL IS
C           DONE BEFORE OPTIMUM THE BUCKET IS SPECIFIED HERE
C          RF IS RL/D RADIUS FOR QUALITY CALCULATION
C
C*****************************************************************************C
```

(3) Namelist THRED -- THRED establishes the parameters required for three-dimensional plotting routines.

```
      NAMELIST / THRED / PLOT3D,TITLE3,DIAM,
     X      PLTISO, RPLOT, DIAISO, PSLICE, NP, JFAZE, XMAG
C
C
C          PLOT3D = .TRUE. FOR THREE DIMENSIONAL PLOTS OF NEAR FIELD
C                 = .FALSE. FOR NO PLOTS
C          TITLE3 = TITLE INFORMATION FOR THREE DIMENSIONAL PLOTS
C          DIAM   = DIAM OF ILLUSTRATED FIELD ON PLOTS
C
C       PLTISO IS LOGICAL FOR ISOPLOTS OF FIELD
C         RPLOT IS THE RADIUS OF CIRCLE DRAWN ON ISOPLOT FOR REFERENCE
C         DIAISO IS DIAMETER OF ISOPLOTS DESIRED
C       PSLICE IS LOGICAL FOR SLICE PLOTS OF FIELD
C         NP = THE SLICE IN Y-DIR. PLOTTED. IF = 0... NP = NPTS/2
C         JFAZE = 0. NO PHASE PLOT FOR THIS
C               = 1. GET THE PHASE
C
C
C   **************************************************************************C
```

(4) Namelist OPTIM - Namelist OPT2 -- These two namelists are used by the optimization portion of the SOQ routine. OPTIM must be read first to direct the optimization procedure. OPT2 establishes which parameters are to be used in the procedure and their constraints.

```
      NAMELIST/ OPTIM / RB,  IPOT, NIND, NBIGIT, DRB
C          RB = BUCKET SIZE FOR QUALITY OPTIMIZATION
C          IPOT = 1 POWER WITHIN RB
C                 2 TOTAL POWER IN BEAM
C                 3 PEAK INTENSITY
C          NIND IS NUMBER OF IND VAIABLES TO BE OPTIMIZED
C          NBIGIT = BIGGEST NUMBER OF ITERATIONS TO BE PERFORMED
C          DRB IS THE BEAM DIAMETER FOR QUALITY CALC...IF CALLED TO QUAL
C            EARLIER THIS IS NOT NEEDED
C
```

```
      NAMELIST/OPT2/ IEL1,   IEL2,  IEL3,  XMIN, XMAX, XADD
C        (IEL1,IEL2,IEL3) IS THE VECTOR DESCRIBING THE POSITION OF THE
C           OPTIMIZED PARAMETER...IN OPERATIONAL SPACE
C        XMIN AND XMAX ARE THE CONSTRAINTS ON THE OPTIMAZED VECTOR
C        XADD IS A CONSTANT ADDED TO THE OPTIMAZED VARIABLE SUCH THAT
C          ITS VALUE IS NEVER EQQUAL TO ZERO
C         THERE ARE NINO NUMBER OF CALLS TO THIS NAMELIST
C
```

(5) Namelist PARAM -- This namelist gives the parameters to be varied and what values are to be used.

```
      NAMELIST / PARAM /  NEL1,NEL2,NEL3,  NPARA,  XNPARA,
     X                    MEL1,MEL2,MEL3,  MPARA,  XMPARA
C        (EL1,EL2,EL3) IS THE VECTOR DESCRIBING THE POSITION OF THE
C         VARIABLE WHICH IS TO BE VARIED
C        NPARA,MPARA ARE THE NUMBER OF CHANGES IN EACH VARIABLE
C        XNPARA,XMPARA ARE THEE ARRAYS THAT CONTAIN THE VALUES WHICH
C         ARE TO BE USED
C *********IF ONLY ONE SET IS TO BE VARIED USE ONLY THE NPARA SET,
C          AND SET  MEL1 = 0,    THE NES ARE THE INNER LOOP  ************
C        IF ANC ARRAY IS TO BE CHANGED AND NO CALL, AT THIS TIME TO
C         AUTO(GOL()), THEN SET NPARA=0...THEN TWO VALUES CAN BE CHANGED
C         IF ONLY ONE IS TO BE CHANGED SET MEL1=0
C ******ALL CALLS BETWEEN GOL AND PARAM TO QUAL,PLOT...WILL BE REPEATED
C         INSIDE THE PARAMETRIC LOOP
C*********************************************************************C
```

(6) Program SOQ (Program SOQ Flow Chart (Fig. 12) appears on page 40.)

```
PROGRAM SOQ           76/176      OPT=1       FIN 4.6+452       04/27/79   12.23.47

      PROGRAM SOQ(OUTPUT,TAPE1,TAPE2,TAPE3,TAPE4,TAPE5,                  CORR1        1
     A   TAPE6=OUTPUT,TAPE7,TAPE8,TAPE9,TAPE10,TAPE11,TAPE12,TAPE13,     MAIN         3
     B   TAPE14,TAPE15,TAPE16,TAPE17,TAPE18,TAPE19,TAPE20,TAPE21,        MAIN         4
     CTAPE22,TAPE23,TAPE24,TAPE25,TAPE26,TAPE27,TAPE28,TAPE29,           SOQ77CY1     1
     DTAPE30,TAPE31)                                                     SOQ77CY1     2
      LEVEL 2,CO,COR,SPACE                                               CORR2        1
      COMMON /FST/ SPACE(1000)                                           CORR2        2
      COMMON/MEL1/CO(16384),CFIL(16512),X(128),WL,NPTS,NPY,DRX,DRY       MAIN         7
      COMMON /PLTSIG/ PLOTSG                                             CROP1        1
      COMMON /INITL/ INI                                                 MAIN         8
      DIMENSION TITLE(20),AS(3),AUP(4),ALOW(4),XUP(4),                   MAIN         9
     AIOP(3,4),XSCR(4),ABC(12,20,9),TITLE3(20),XUPADD(4),                CIOASTG      1
     2  XMPARA(10),    XNPARA(10),  MAINE(25), TITLES(20), COR(32768)    MAIN        11
      COMPLEX CO,CFIL,CORS                                               MAIN        12
      LOGICAL RESTRT,PLOT3D,PLTISO,PSLICE,CALQL,SYMTRC                   MAIN        13
      EQUIVALENCE (CO(1),COR(1))                                         MAIN        14
      DATA WWL,DCAL,NNPTS,DDRX,DDRY/-1.,0.,0,0.,0./                      CORR2        3
      DATA DCAL,RESTRT,IN,IB,NCALL,AMPGES,ITNUM,SYMTRC,DGAUSS,PHIRAD     MAIN        15
     X   / 0.0, .TRUE., 0, 8,  2,     1.0,   -1, .FALSE.,  0.0 , 0.0   / MAIN        16
      DATA TITLES/20*4H    /                                             MAIN        17
      DATA IQLT,DB,ISAV,IPHASE,RBH,RF /0,0,0,0,3,1.0,8.0/                MAIN        18
      DATA TITLE/20*4H    /                                              MAIN        19
      DATA RH,IPUT,NIND,NDIGIT,DBB /2.0,1,0,1,0.0/                       MAIN        20
      DATA PLOT3D,DIAM,PLTISO,RPLOT,DIAISO,PSLICE,NP,JFAZE,XMAG          MAIN        21
     X   /.FALSE.,0.0, .FALSE.,0.0, 0.0,    .FALSE.,0, 0,1.0/            MAIN        22
      DATA PLOTS / 0.  /                                                 CROP1        2
      DATA TITLE3/20*4H    /                                             MAIN        23
```

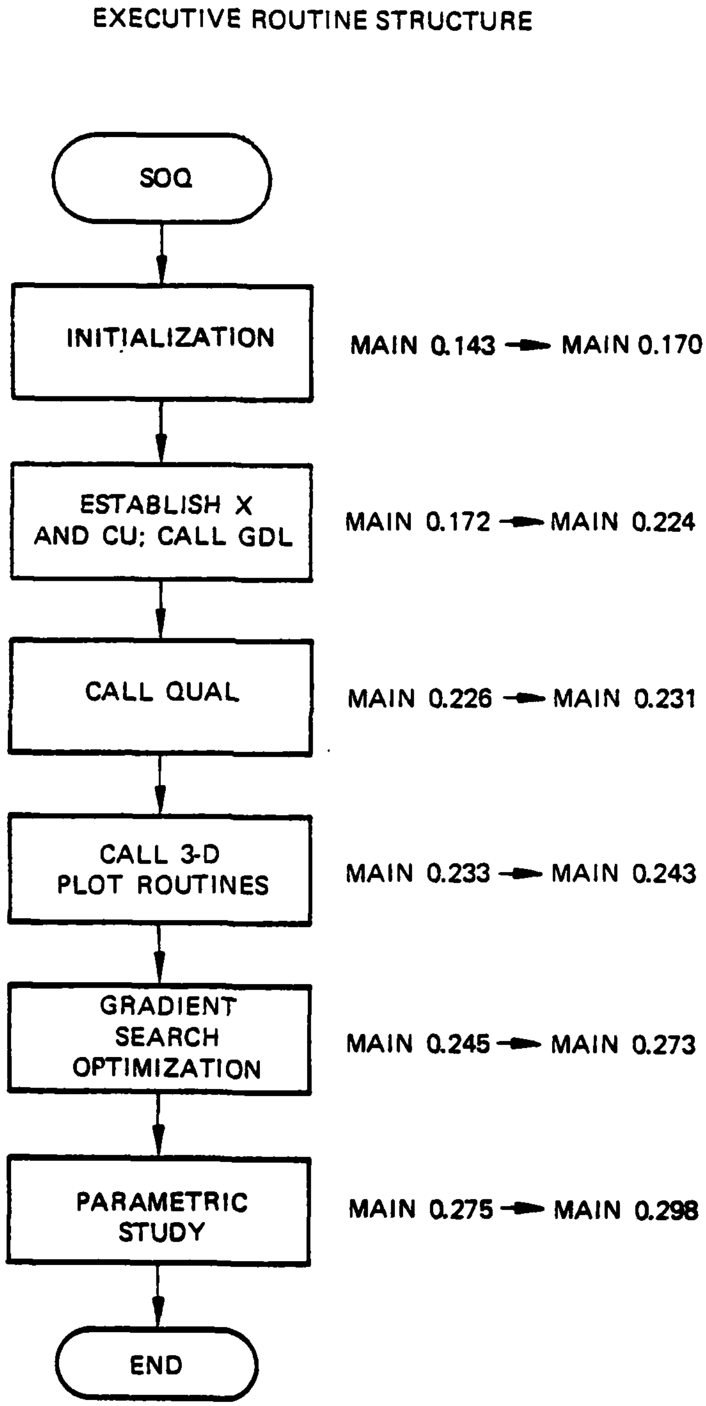


Figure 12. Program SOQ flow chart.

```
C*****************************************************************C
      NAMELIST /START/ WWL,OCAL,NNPTS,DDRX,DDRY,RESTRT,IN,IB,NCALL,
     X AMPGES,  ITNUM,  SYMTRC  ,DGAUSS , TITLES, PHIRAD
     X  , PLOTS
C PLOTS=-1.,  AMPLITUDE , PHASE SLICE PLOTS
C PLOTS= 0.,  NO SLICE OR ISO-INTENSITY PLOTS
C PLOTS= 1.,  INTENSITY, PHASE SLICE PLOTS
C PLOTS=1., INTENSITY, PHASE SLICE PLOTS          OT(IPLTS)
C       NCALL CONTROLS THE MOVEMENT INSIDE MAIN
C         = 2, GOL SECTION,CALLS GOL AND READ CU+X FROM DISK
C         = 3 CALL TO QUALITY ALGORYTHUM, READS QLOT
C         = 4 CALL TO ANY OF THE GOULD PLOTTING PACKAGES, READS THRED
C         = 5 STARTS OPTIMAZATION ALA DAVIDN, READS OPTIM
C         = 6 PARAMETRIC STUDIES..INVOLVES CHANGING ABC ARRAY FOR GOL,
C             READS PARAM
C*****************************************************************C
C      WL IS RADIATION WAVELENGHT
C      OCAL IS INTIAL SIZE OF CALCULATION REGION
C      NPTS IS NUMBER OF FIELD POINTS ACROSS OCAL
C      DRX,DRY = THE (X,Y) POSITION OF THE CENTER OF CU RELATIVE
C                TO THE OPTICAL AXIS
C      RESTRT IS CONTROL FOR RESTARTING WITH EXISTING GAIN CO-EFF AND
C      INITIAL FIELD FROM PREVIOUS RUN.........
C          .TRUE. IF RESTARTING, OR IF FIELD IS TO BE READ FROM IB
C          .FALSE. IF NOT, OR IF INITIAL FIELD AND XES ARE TO BE CALC
C      IN = UNIT NUMBER OF DATA SET FOR GOL AND CAVITY
C         IF IN = 0, THEN THERE IS NO CALL TO GOL
C
C      IB IS UNIT NUMBER OF INPUT FIELD TO GOL
C        IF IB = 0, THEN NOTHING IS READ
C      AMPGES IS INITIAL AMPLITUDE OF STARTING BEAM(PEAK AMPLITUDE
C      FOR GAUSSIAN)
C      PHIRAD IS IS PHASE FRONT RADIUS OF CURVATURE (=0.0 FOR PLANE)
C      ITNUM IS THE ITERATION NUMBER...IF UNSPECIFIED IT READS OFF DISK
C
C         SYMTRC IS LOGICAL FOR SYMMETRIC ANAYLSYS OR NOT
C      DGAUSS IS DIAMETER AT WHICH GAUSSIAN AMPLITUDE = 1.0/E
C
C*****************************************************************C
C
C
      NAMELIST/QLOT/ TITLE,  IQLT, DB, ISAV, IPHASE, RBB ,RF
C
C      TITLE FOR PLOTS IN QUALITY ROUTINE
C      IQLT IS PLOTTING PARAMETER FOR PLTOT....QUALITY PLOTS
C          = 0 ISO-INTENSITY AND POWER VS RL/D GOULD PLOTS
C          = 1 FAR-FIELD PWR VS RL/D GOULD.......
C          = 2 NO GOULD PLOTS  CALCULATES POWER DIST. ONLY
C          = 3 ISO-INTENSITY GOULD, PWR DIST, BUT NO PWR RL/D PLOTS
C          = 4 CALC POWER INSIDE RBB ONLY...NO CALL TO PLTOT
C      DB = BEAM DIAMETER
C      ISAV IS SAVING PARAMETER........
C          =0, DONT SAVE        =1 SAVE INPUT FIELD
C          =-1 USE DATA SET #9 FOR INPUT
C      IPHASE CONTROLS THE PHASE CORRECTIONS APPLIED TO THE FIELD
C          = 0  NONE
C          = 1  PLANAR CORRECTION
C          = 2  SPHERICAL
C          = 3  BOTH
C         RBB IS THE BUCKET SIZE FOR OPTIMAZATION...IF A CALL TO QUAL IS
C           DONE BEFORE OPTIMUM THE BUCKET IS SPECIFIED HERE
C         RF IS RL/D RADIUS FOR QUALITY CALCULATION
C
C*****************************************************************C
C
C
      NAMELIST/ OPTIM / RB,  IPOT, NIND, NBIGIT, DBB
C        RB = BUCKET SIZE FOR QUALITY OPTIMIZATION
C        IPOT = 1 POWER WITHIN RB
C               2 TOTAL POWER IN BEAM
C               3 PEAK INTENSITY
C         NIND IS NUMBER OF IND VAIABLES TO BE OPTIMIZED
```

```
C        NBIGIT = BIGGEST NUMBER OF ITERATIONS TO BE PERFORMED
C        DBB IS THE BEAM DIAMETER FOR QUALITY CALC...IF CALLED TO QUAL
C          EARLIER THIS IS NOT NEEDED
C
      NAMELIST/OPT2/ IEL1,   IEL2,  IEL3,  XMIN, XMAX, XADD
C       (IEL1,IEL2,IEL3) IS THE VECTOR DESCRIBING THE POSITION OF THE
C          OPTIMIZED PARAMETER...IN OPERATIONAL SPACE
C       XMIN AND XMAX ARE THE CONSTRAINTS ON THE OPTIMAZED VECTOR
C       XADD IS A CONSTANT ADDED TO THE OPTIMAZED VARIABLE SUCH THAT
C         ITS VALUE IS NEVER EQQUAL TO ZERO
C        THERE ARE NINO NUMBER OF CALLS TO THIS NAMELIST
C
C
C*****************************************************************C
C
      NAMELIST / PARAM /  NEL1,NEL2,NEL3,  NPARA,  XNPARA,
     X                    MEL1,MEL2,MEL3,  MPARA,  XMPARA
C       (EL1,EL2,EL3) IS THE VECTOR DESCRIBING THE POSITION OF THE
C        VARIABLE WHICH IS TO BE VARIED
C       NPARA,MPARA ARE THE NUMBER OF CHANGES IN EACH VARIABLE
C       XNPARA,XMPARA ARE THEE ARRAYS THAT CONTAIN THE VALUES WHICH
C        ARE TO BE USED
C ********IF ONLY ONE SET IS TO BE VARIED USE ONLY THE NPARA SET,
C          AND SET  MEL1 = 0.   THE NES ARE THE INNER LOOP  ************
C       IF ABC ARRAY IS TO BE CHANGED AND NO CALL, AT THIS TIME TO
C        AUTO((GOL)), THEN SET NPARA=0...THEN TWO VALUES CAN BE CHANGED
C         IF ONLY ONE IS TO BE CHANGED SET MEL1=0
C *****ALL CALLS BETWEEN GOL AND PARAM TO QUAL,PLOT...WILL BE REPEATED
C        INSIDE THE PARAMETRIC LOOP
C*****************************************************************C
C
C
      NAMELIST / THREO / PLOT3D,TITLE3,DIAM,
     X      PLTISO, RPLOT, DIAISO, PSLICE, NP, JFAZE, XMAG
C
C
C        PLOT3D = .TRUE. FOR THREE DIMENSIONAL PLOTS OF NEAR FIELD
C               = .FALSE. FOR NO PLOTS
C        TITLE3 = TITLE INFORMATION FOR THREE DIMENSIONAL PLOTS
C        DIAM   = DIAM OF ILLUSTRATED FIELD ON PLOTS
C
C     PLTISO IS LOGICAL FOR ISOPLOTS OF FIELD
C       RPLOT IS THE RADIUS OF CIRCLE DRAWN ON ISOPLOT FOR REFERENCE
C       DIAISO IS DIAMETER OF ISOPLOTS DESIRED
C     PSLICE IS LOGICAL FOR SLICE PLOTS OF FIELD
C       NP = THE SLICE IN Y-DIR. PLOTTTED, IF = 0... NP = NPTS/2
C      JFAZE = 0, NO PHASE PLOT FOR THIS
C             = 1, GET THE PHASE
C
C
C  ***************************************************************C
      CALL LIST80(5)
      INT=0
      ICNVRG=0
      WL=-1.
      DRX = 0.
      DRY = 0.
      PI=3.141592
      DO 14 II=1,4
   14 XSCH(II)=1.
      IRRK = 1
      MAINE(1) = 1
      INOLD=5
      RF = 8.
  999 READ(5,START)
      WL = WWL
      NPTS=NNPTS
      DRX=DDRX
      DRY=DDRY
      PLOTSG = PLOTS
      READ (5,1243) TITLES
 1243 FORMAT (20A4)
```

```
      IF (WL .LE. 0.) GO TO 9876
      WRITE(6,150) TITLES
  150 FORMAT(1H1,30(3HSOQ)/1X,1HO,88X,1HO/1X,1HO,4X,20A4,4X,1HO/
     X 1X,1HO,88X,1HO/1X,30(3HSOQ)////)
      NPY =  NPTS
      IF (SYMTRC)  NPY = NPY/2
      NOB = NPTS * NPY
      NOKY = 0
      ABC(1,2,1) = DRX
      ABC(2,2,1) = DRY
      IMRK = IMRK + 1
      MAINE(IMRK) = NCALL
      GO TO (999,100,200,300,400,500),NCALL
C   ****************************************************************
C   TRANSFER CONTROL TO GOL
  100 IF ( RESTRT .OR. IB.EQ.0 )  GO TO 3
      DX=DCAL/NPTS
      X(1)=-DCAL/2.+DX/2.
      DO 2 I=2,NPTS
    2 X(I)=X(I-1)+DX
      DO 9 I=1,NOB
    9 CU( I ) = CMPLX(AMPGES,0.)
      IF (PHIRAD.EQ.0.)GO TO 71
      ROFACT=-PI/(WL*PHIRAD)
      DO 72 J=1,NPY
      J1=(J-1)*NPTS
      YSQ = X(J)**2
      DO 72 I=1,NPTS
      KKK=J1+I
      KKK2= 2 * KKK
      KKK2M1 = KKK2 - 1
      PHI =  ROFACT * (X(I)**2+YSQ)
      SINP = SIN(PHI)
      COSP = COS(PHI)
      CORS = COR(KKK2M1)
      COR(KKK2M1) = CORS*COSP - COR(KKK2)*SINP
   72 COR(KKK2)    = CORS*SINP + COR(KKK2)*COSP
   71 IF(DGAUSS.EQ.0.) GO TO 50
      SIGMA=DGAUSS**2/4.0
      DO 51 J=1,NPY
      NROW=(J-1)*NPTS
      YSQ = X(J)**2
      DO 51 I=1,NPTS
      NCNT=NROW+I
      CU(NCNT)=CU(NCNT)*EXP(-(X(I)**2+YSQ)/SIGMA)
   51 CONTINUE
      WRITE(6,52)DGAUSS,AMPGES
   52 FORMAT(85H0   GAUSSIAN AMPLITUDE DISTRIBUTION HAS BEEN FORMED WITH
     X A 1/E AMPLITUDE AT DIAMETER=,F10.2/18H    PEAK AMPLITUDE=,G12.5/)
   50 CONTINUE
      NIT = 0
      GO TO 4
    3 IF (IB.EQ.0) GO TO 4
      READ(IB) (CU(I),I=1,NOB),X,DR1,DR2,NIT
      REWIND IB
    4 IF (IN .EQ. 0) GO TO 999
      IF (IN.EQ.INOLD.OR.IN.EQ.5.) GO TO 5
      WRITE (6,6) IN
    6 FORMAT (27H1  THE INPUT DATA ON SET # ,I2,21H FOR THIS CALL TO GOL
     X/)
      CALL LISTER(IN)
      INOLD=IN
    5 IF (ITNUM .GE. 0)  NIT = ITNUM
      CALL GOL(IN,RESTRT,ABC,NIT,IB,0)
      MOLD = IMRK
      CALQL = .FALSE.
      GO TO 999
C   ****************************************************************
C   TRANSFER CONTROL TO QUAL
  200 READ(5,QLD1)
      READ (5,1203) TITLE
      CALQL = .TRUE.
```

```
  210 CALL QUAL  (IPHASE,ISAV,IQLT,TITLE,RHB,AS,DB,RF)
      GO TO 997
C     *************************************************************
C     TRANSFER CONTROL TO PLOTTING ROUTINES
  300 READ(5,THRED)
      IF(XMAG.EQ.1.)GO TO 310
      DO 377 IMG=1,NPTS
  377 X(IMG)=X(IMG)*XMAG
      DO 378 IMG=1,NOB
  378 CO(IMG)=CO(IMG)/XMAG
  310 IF (PLOT3D) CALL NEAR(DIAM,TITLE3)
      IF (PLTISO) CALL ISOS(TITLE3,RPLOT,DIAISO)
      IF (PSLICE) CALL PRETYP(NP,TITLE3,JPAZE)
      GO TO 997
C     *************************************************************
C     PERFORM GRADIENT SEARCH OPTIMIZATION
  400 DO 8 II=1,NIND
      READ (5,OPT2)
      IOP(1,II) = IEL1
      IOP(2,II) = IEL2
      IOP(3,II) = IEL3
      XLOW(II) = XMIN
      XUP(II) = XMAX
      XOPADD(II) = XADD
    8 XOP(II) = ABC(IOP(1,II),IOP(2,II),IOP(3,II))+XOPADD(II)
  410 IF (.NOT.CALQL) CALL QUAL(0,0,4,TITLE,RB,AS,DBB)
   32 QQQ  = 1./ AS(IPOT)
      CALL CNSTRN (XOP,XLOW,XUP,NIND,QQQ,QQQQ)
      CALL DAVION (QQQQ,XOP,NIND,ICNVRG,NBIGIT,.002,0.,XSCH)
      GO TO (919,918),ICNVRG
      DO 13 II=1,NIND
   13 ABC(IOP(1,II),IOP(2,II),IOP(3,II)) = XOP(II) -XOPADD(II)
C     CALL AUTO(ABC,IB)
      CALL GOL(IN,RESTRT,ABC,NIT,IB,1)
      IF (MOLD.EQ.IMRK-1) GO TO 410
      NOKY = MOLD
      GO TO 997
  919 WRITE (6,23) (ABC(IOP(1,IW),IOP(2,IW),IOP(3,IW)),IW=1,NIND), AS
   23 FORMAT (82H1     OPTIMAZATION ROUTINE HAS CHOSEN THE FOLLOWING PARA
     XMETERS...AND THIS MAXIMUM  /5F12.5//)
      GO TO 999
  918 WRITE (6,24)
   24 FORMAT (52H1  OPTIMUM ROUTINE HAS EXCEEDED MAX # OF ITERATIONS )
      GO TO 999
C     *************************************************************
C     PERFORM PARAMETRIC STUDY
  500 READ(5,PARAM)
      JJ1 = 0
      IF (NPARA.NE.0) GO TO 510
      IMRK = IMRK - 1
      ABC(NEL1,NEL2,NEL3) = XNPARA(1)
      IF (MPARA.EQ.0) ABC(MEL1,MEL2,MEL3) = XMPARA(1)
      GO TO 999
  510 JJ2 = 0
      IF (MEL1.EQ.0) MPARA = 1
      JJ1 = JJ1 + 1
      IF (JJ1.GT.MPARA) GO TO 999
      IF (MEL1  .NE. 0)
     X      ABC(MEL1,MEL2,MEL3) = XMPARA(JJ1)
  520 JJ2 = JJ2 + 1
      IF (JJ2 .GT. NPARA) GO TO 510
      ABC(NEL1,NEL2,NEL3) = XNPARA(JJ2)
      CALL GOL(IN,RESTRT,ABC,NIT,IB,1)
C     CALL AUTO(ABC,IB)
      NOKY = MOLD
      GO TO 997
  997 NOKY = NOKY + 1
      NNN = MAINE(NOKY)
      GO TO (999,100,210,310,410,520), NNN
 9876 STOP
      END
```

2. SUBROUTINE LIST80

Calls: N/A

Called by: MAIN

Subroutine LIST80 is called by the executive routine MAIN to list data input to the SOQ code. The LIST80 flow chart (Fig. 13) appears on page 45.

After control is passed to LIST80, header information is printed. The input unit is read and a counter, KARD, is incremented for each record read. The input data is reformatted and printed on the line printer. When an end-of-file is received from the input unit, it is backspaced K records and control is returned to MAIN.

<u>Arguments</u>

K Unit number on which input is read (usually 5).

<u>Relevant Variables</u>

C Card inputs read and printed as read.

SUBROUTINE LIST80 76/176 OPT=1 FIN 4.6+452 04/27/79 12.23.47

```
      SUBROUTINE LIST80(K)
C         THIS ROUTINE WRITES NAMELIST INPUT HOPEFULLY
      DIMENSION C(20)
      WRITE(6,35)
      KARD = 1
   30 WRITE(6,20)
   20 FORMAT(4(/),1X,52(1H*),12H*CARD INPUTS, 52(1H*))
      WRITE(6,40)
   40 FORMAT(/,
     *5H*CARD,15X,10(1H1),10(1H2),10(1H3),10(1H4),10(1H5),
     *10(1H6),10(1H7),1H8,/7HSCOLUMN,4X,8(10H1234567890),5X,
     *5HCARD,/)
      DO 25 J = 1,45
    1 READ(K,5) C
      IF (EOF(K).NE.0.0)GO TO 15
    5 FORMAT(20A4)
      WRITE(6,10) C,KARD
   10 FORMAT( 10X,20A4,I8)
      KARD = KARD + 1
   25 CONTINUE
      WRITE(6,40)
      WRITE(6,35)
      GO TO 30
   15 IBACK = KARD - 1
      DO 45 I = 1,IBACK
   45 BACKSPACE K
      WRITE(6,40)
      WRITE(6,35)
   35 FORMAT(1H1)
       RETURN
       END
```

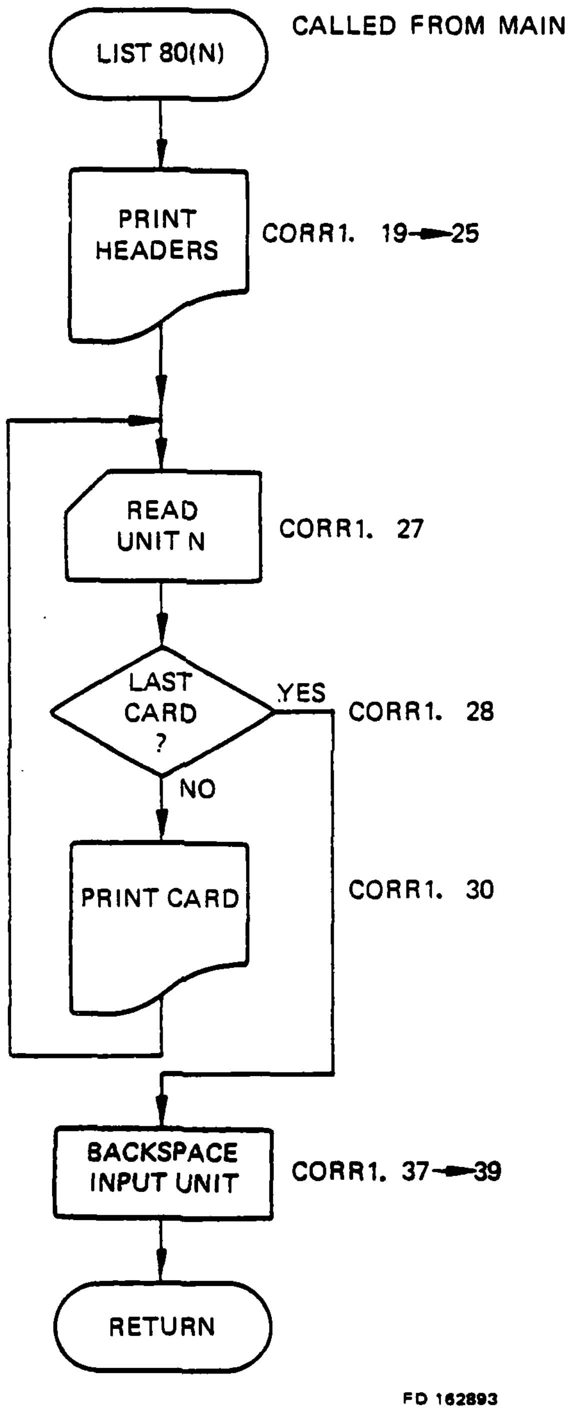


Figure 13. Subroutine LIST80 flow chart.

3. SUBROUTINE AEROW

Subroutine AEROW is used to apply a random phase variation to the complex field. Figure 14 shows the subroutine AEROW flow chart.

AERO is entered with the complex field array real coefficients, CUR, and with the number of points in x and y.

SIGMAM is a constant established by previous aerowindow work. It is later multiplied by the random number returned from the RANDU call to give the proper random phase range for an aerowindow.

Inside the DO LOOP, the random phase is obtained and the sine and cosine of the negative of this phase is taken. A negative number is required to yield a diverging phase impact.

The complex field, CU, is represented by a complex number, a + ib, whereas the CUR variables represent the real coefficients alone.

$$CU(1) \quad \begin{cases} CUR(1) = a \\ CUR(2) = b \end{cases} \quad a + ib \tag{11}$$

The random phase is applied by:

$$\overbrace{(a + ib)}^{CU} \quad * \varepsilon^{i\phi} \tag{12}$$

$$(a + ib)\ (\cos\phi + i\sin\phi) \tag{13}$$

$$\left.\begin{aligned} a\cos\phi - b\sin\phi &\rightarrow CUR\ (1) \\ b\cos\phi + a\sin\phi &\rightarrow CUR\ (2) \end{aligned}\right\} \quad CU(1) \tag{14}$$

<u>Argument List</u>

CUR	Complex field array
NPTS	Number of x points
NPY	Number of y points

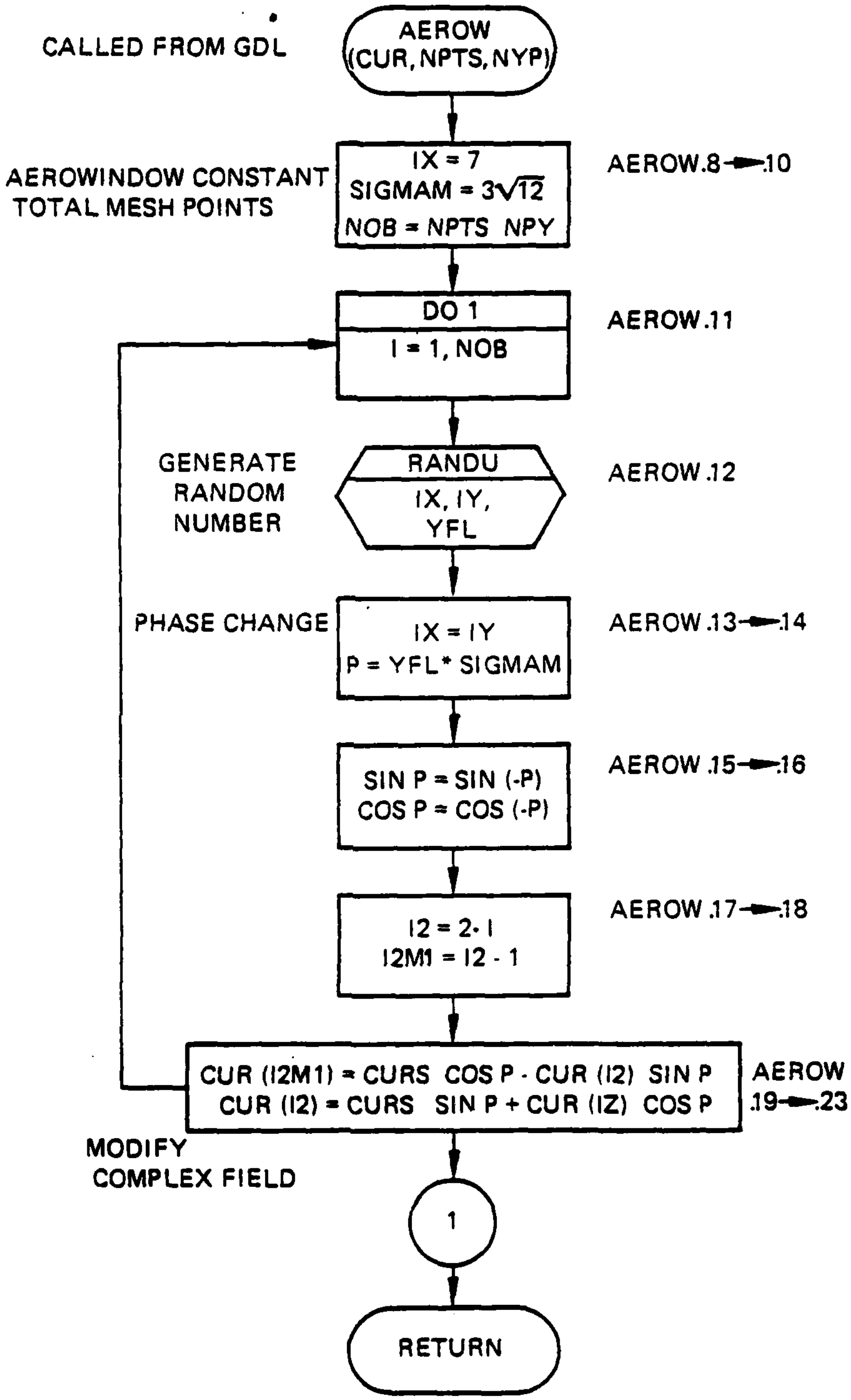


Figure 14. Subroutine AEROW flow chart.

Relevant Variables

CURS	Odd number members of field CUR
P	Phase change
SIGMAM	Aerodynamic window constant = $0.3\sqrt{12}$
YFL	Random number generated by "RANDU"

SUBROUTINE AEROW   76/176   OPT=1   FIN 4.6+452   04/27/79   12.23.47

```
      SUBROUTINE AEROW (CUR,NPTS,NPY)                               AEROW
C     AERODYNAMIC WINDOW MODEL                                      AEROW
C     THIS ROUTINE APPLIES A RANDOM PHASE VARIATION TO THE COMPLEX  AEROW
C     FIELD                                                         AEROW
      LEVEL 2, CUR,NPTS,NPY                                         AEROW
      DIMENSION CUR(1)                                              AEROW
      IX = 7                                                        AEROW
      SIGMAM= 0.300 * SQRT(12.)                                     AEROW
      NOB = NPTS*NPY                                                AEROW
      DO 1 I = 1,NOB                                                AEROW
      CALL RANDU (IX,IY,YFL)                                        AEROW
      IX = IY                                                       AEROW
      P = YFL * SIGMAM                                              AEROW
      SINP = SIN(-P)                                                AEROW
      COSP = COS(-P)                                                AEROW
      I2   = 2*I                                                    AEROW
      I2M1 = I2 - 1                                                 AEROW
      CURS = CUR(I2M1)                                              AEROW
      CUR(I2M1) = CURS*COSP - CUR(I2)*SINP                          AEROW
    1 CUR(I2)   = CURS*SINP + CUR(I2)*COSP                          AEROW
C   1 CU( I ) =CU( I ) * CEXP(CMPLX(0.,-P))                         AEROW
      RETURN                                                        AEROW
      END                                                           AEROW
```

4. SUBROUTINE RANDU

Subroutine called by AEROW returns rectangularly distributed random numbers in the range 0 to 1 in the variable YFL. Figure 15 shows the RANDU flow chart.

SUBROUTINE RANDU   76/176   OPT=1   FIN 4.6+452   04/27/79   12.23.47

```
      SUBROUTINE RANDU  (IX,IY,YFL)                                 RANDU
C     RANDOM NUMBER GENERATOR                                       RANDU
C     THIS ROUTINE SUPPLIES THE RANDOM NUMBERS TO AEROW             RANDU
      IY = IX*899                                                   RANDU
      IF (IY) 5,5,6                                                 RANDU
    5 IY = IY+ 2147483647 + 1                                       RANDU
    6 YFL = IY                                                      RANDU
      YFL = YFL/2147483647.                                         RANDU
      RETURN                                                        RANDU
      END                                                           RANDU
```

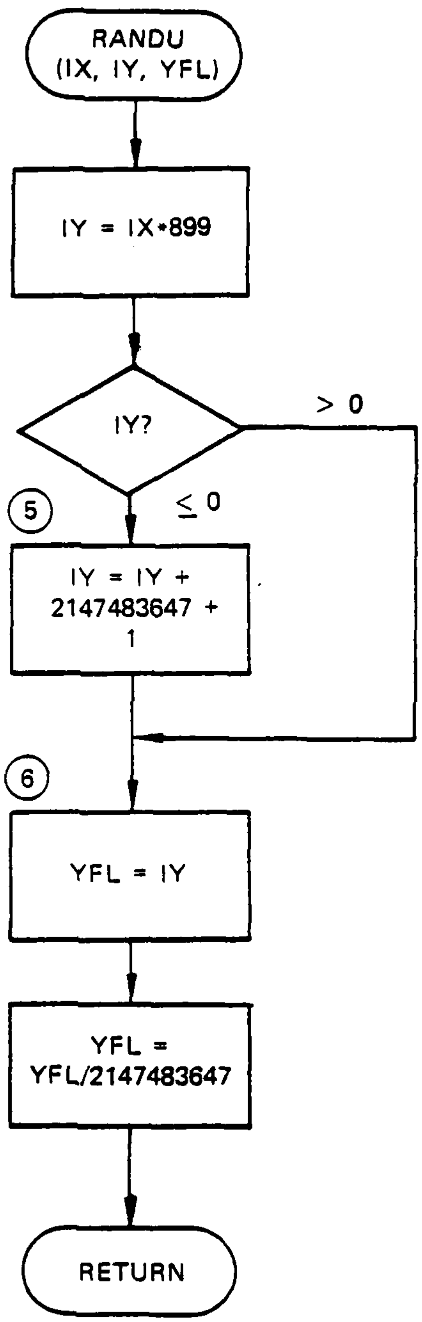


Figure 15. Subroutine RANDU flow chart.

5. SUBROUTINE APRTR

Called by: MIRROR, GDL

Calls: N/A

a. Purpose -- Subroutine APRTR applies an aperture, either circular or rectangular (Fig. 16), with or without a central obscuration, to the complex field. It also determines the value and position of maximum intensity on the aperture plate. Figure 17 shows the APRTR flow chart.

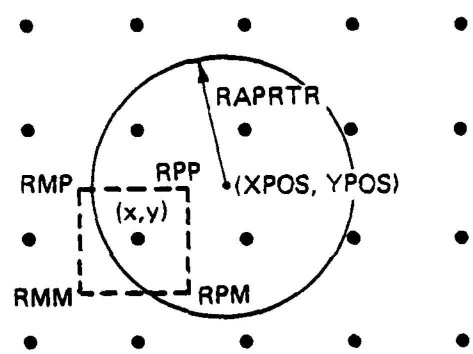


Figure 16. Subroutine APRTR nomenclature.

APRTR is entered with the inner and outer obscuration dimensions along with the coordinates of the aperture.

A test is made to see if the aperture is rectangular or circular. The appropriate boundary parameters are computed. Each point in the complex field is checked to see if it will pass through the clear aperture. If so, it is left alone. If not, it is zeroed out after it has been checked to determine if it is the location of maximum intensity on the aperture plate.

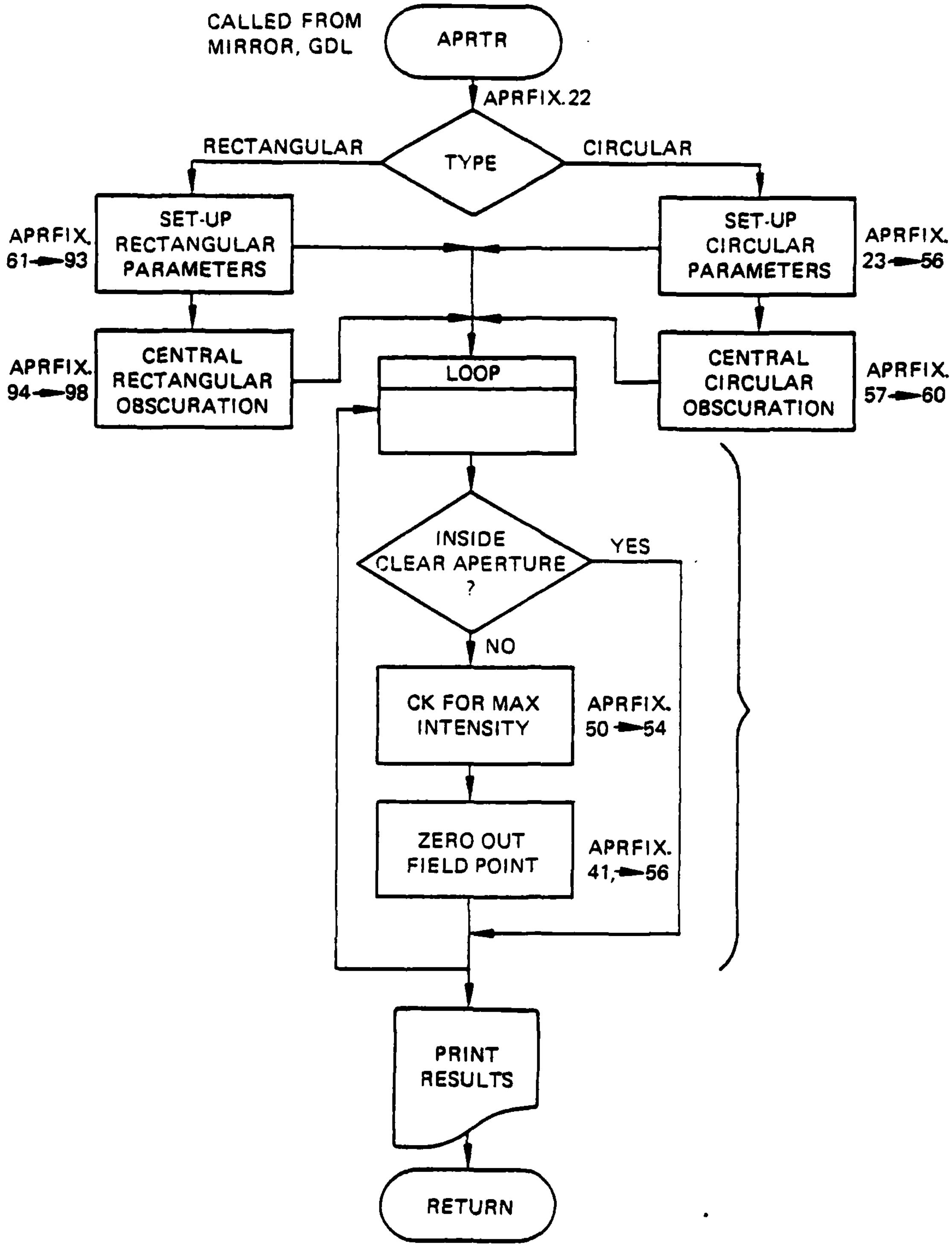


Figure 17. Subroutine APRTR flow chart.

The transmission function is

$$t(x, y) = \begin{cases} \text{RDISK} \leq \sqrt{(x\text{-xpos})^2 + (y\text{-ypos})} \leq \text{RAPRTR} \\ 0 \text{ otherwise} \end{cases} \tag{15}$$

b. Relevant formalism

$$\text{RPP} = \left(|x| + \frac{dx}{2}\right)^2 + \left(|y| + \frac{dy}{2}\right)^2 \tag{16}$$

$$\text{RMM} = \left(|x| - \frac{dx}{2}\right)^2 + \left(|y| - \frac{dy}{2}\right)^2 \tag{17}$$

$$\text{RMP} = \left(|x| - \frac{dx}{2}\right)^2 + \left(|y| + \frac{dy}{2}\right)^2 \tag{18}$$

$$\text{RPM} = \left(|x| + \frac{dx}{2}\right)^2 + \left(|y| - \frac{dy}{2}\right)^2 \tag{19}$$

These four locations represent an area surrounding the particular point of interest as shown in Figure 16. For each of these sets of points the locations of the aperture and obscuration are checked. If all the four points impinge on an aperture or central obscuration, then the intensity at that location is computed and checked for maximum value, then the field is zeroed out (by the impingement).

$$\text{Int} = (\text{ReCu})^2 + (\text{ImCu})^2 \tag{20}$$

$$\text{Max Int} = \text{AMAX (Int, Max Int)} \tag{21}$$

$$\text{PER} = 0 \tag{22}$$

$$\text{Cu} = \text{CU X PER} \tag{23}$$

If all four points lie within the clear aperture, the field is unchanged.

$$PER = 1 \tag{24}$$

$$CU = CU \times PER \tag{25}$$

If the four points encompass an aperture edge, then the intensity is prorated on a percentage basis and transmitted.

$$PER = (RAD\text{-}RMIN) \,/\, RMAX\text{-}RMIN) \tag{26}$$

$$CU = CU \times PER \tag{27}$$

where

$$RMAX = MAX \text{ of } (RPP,\ RMM,\ RMP,\ RPM) \tag{28}$$

$$RMIN = MIN \text{ of } (RPP,\ RMM,\ RMP,\ RPM) \tag{29}$$

$$RAD = \text{Radius (or x or y dimension) at aperture edge} \tag{30}$$

Argument List

RAPRTR Radius of circular aperture (cm) or x-dimension (half width) of rectangular aperture (cm)

RDISK Radius of central obscuration of a circular aperture (cm); or x-dimension (half width) of a rectangular central obscuration

XPOS x location of aperture center with respect to optic center-line (cm)

YPOS y location of aperture center with respect to optic center-line (cm)

YAPRTR y dimension (half height) of rectangular aperture (cm)

YDISK y dimension (half height) of a rectangular central obscuration (cm).

Relevant Variables

A	Half width of rectangular aperture (cm)
AINT	Intensity (W/cm^2)
AINTMX	Maximum intensity (W/cm^2)
B	Half height of rectangular aperture (cm)
DX	x distance between points in the mesh (cm)
DY	y distance between points in the mesh (cm)
RAD	= RAPRTR, aperture radius (cm)
X	x location adjusted for centerline difference and accumulated dx (cm)
XAR (N)	x or y position of N (cm)
Y	y location adjusted for centerline difference and accumulated dy (cm).

Commons Modified

/MELT/

Array modified CU(I) @ APRFIX.56,93.

SUBROUTINE APRTR 76/176 OPT=1 FIN 4.6+452 04/27/79 12.23.47

```
      SUBROUTINE APRTR (RAPRTR, RDISK, XPOS, YPOS, YAPRTR, YDISK)          APRFIX     1
C       APERTURE MODEL                                                     APRFIX     2
C       THIS ROUTINE APPLIES A CIRCULAR APERTURE WITH RADIUS = RAPRTR      APRFIX     3
C       AND A CENTRAL OBSCURATION WITH RADIUS = RDISK CENTERED ABOUT       APRFIX     4
C       XPOS, YPOS.                                                        APRFIX     5
C ((    MODIFIED (4/8/76 F. ADAMEK) FOR RECTANGULAR APERTURE               APRFIX     6
C       OF WIDTH = 2* RAPRTR, HEIGHT= 2* YAPRTR AND A CENTRAL              APRFIX     7
C ))    OBSCURATION OF WIDTH = 2* RDISK, HEIGHT = 2* YDISK                 APRFIX     8
C (     CAPABILITY FOR FINDING VALUE AND POSITION OF MAX INTENSITY ON      APRFIX     9
C )     APERTURE PLATE ADDED 4/12/76 FAA.                                  APRFIX    10
        LEVEL 2,CU                                                         APRFIX    11
        COMMON/MELT/CU(16384),CFIL(16512),XAR(128),WL,NPTS,NPY,DRX,DRY     APRFIX    12
        COMMON/WAY/WNOW,NREG,RAPTR                                         APRFIX    13
        COMPLEX CU,CFIL                                                    APRFIX    14
        RD(XX,YY,IX,IY)=SQRT((ABS(XX)+IX*DX/2.)**2+(ABS(YY)+DY/2.*IY)**2)  APRFIX    15
        DX=XAR(2)-XAR(1)                                                   APRFIX    16
        DY=DX                                                              APRFIX    17
        IIN=0                                                              APRFIX    18
C ((                                                                       APRFIX    19
        INTCK=0                                                            APRFIX    20
        AINTMX=0.                                                          APRFIX    21
        IF(YAPRTR.NE.0..OR.YDISK.NE.0.) GO TO 180                          APRFIX    22
C )) ********* CIRCULAR APERTURE *************                             APRFIX    23
        WRITE(6,1000) RAPRTR,RDISK                                         SQAPR      1
        IF(RAPRTR.EQ.0.0)GO TO 160                                         APRFIX    24
        RAD=RAPRTR                                                         APRFIX    25
     99 DO 101 IIX=1,NPTS                                                  APRFIX    26
        X=XAR(IIX)+DRX-XPOS                                                APRFIX    27
        DO 101 IIY=1,NPY                                                   APRFIX    28
        Y=XAR(IIY)+DRY-YPOS                                                APRFIX    29
```

```
C (
      R  =SQRT(X**2+Y**2)
      IF (R.GE.RAPRTR) INTCK=1
C )
      RPP=RO(X,Y,1,1)
      RMM=RO(X,Y,-1,-1)
      RMP=RO(X,Y,-1,1)
      RPM=RO(X,Y,1,-1)
      PER=1.
      RMAX=AMAX1(RPP,RMM,RMP,RPM)
      IF (RMAX.LE.RAD) GO TO 100
      PER=0.
      RMIN=AMIN1(RPP,RMM,RMP,RPM)
      IF (RMIN.GE.RAD) GO TO 100
      PER=(RAD-RMIN)/(RMAX-RMIN)
  100 IF (IIN.EQ.1) PER=1.-PER
      NNN = IIX+(IIY-1)*NPTS
C (
      IF(INTCK.EQ.0) GO TO 101
      INTCK=0
      AINT=REAL(CU(NNN))**2 + AIMAG(CU(NNN))**2
      AINTMX=AMAX1(AINT,AINTMX)
      IF (AINT.NE.AINTMX) GO TO 101
      XINTMX=X
      YINTMX=Y
C )
  101 CU(NNN) = CU(NNN) * SQRT(PER)
  160 IF (RDISK.EQ.0..OR.IIN.EQ.1) GO TO 300
      IIN=1
      RAD=RDISK
      GO TO 99
C (( ********* RECTANGULAR APERTURE **************
  180 CONTINUE
      RO=2.*RAPRTR
      WO=2.*YAPRTR
      RI=2.*RDISK
      WI=2.*YDISK
      WRITE(6,1001) RO,WO,RI,WI
 1000 FORMAT(/*   CIRCULAR APERTURE APPLIED*//*   OUTSIDE RADIUS =*,G8.3
     X,/*    INSIDE RADIUS =*,G8.3//)
 1001 FORMAT(/*   RECTANGULAR APERTURE APPLIED*//*   OUTSIDE DIMENSIONS
     XARE *,G8.3,* HIGH BY *,G8.3,* WIDE*/*   INSIDE DIMENSIONS ARE *,
     X G8.3,*HIGH BY *,G8.3,* WIDE*/)
      IF(RAPRTR.EQ.0.0) GO TO 260
      A =RAPRTR
      B =YAPRTR
  199 DO 201 IIX=1,NPTS
      X=XAR(IIX)+DRX-XPOS
      DO 201 IIY=1,NPY
      Y=XAR(IIY)+DRY-YPOS
C (
      IF (ABS(X).GE.RAPRTR.OR.ABS(Y).GE.YAPRTR) INTCK=1
C )
      XMIN = ABS(X)-DX/2
      XMAX = ABS(X)+DX/2
      YMIN = ABS(Y)-DY/2
      YMAX = ABS(Y)+DY/2
      PER=0.
      IF (XMIN.GE.A.OR.YMIN.GE.B) GO TO 200
      PER=1.
      IF (XMAX.LE.A.AND.YMAX.LE.B) GO TO 200
      IF (XMAX.GE.A) PER=(A-XMIN)/DX
      IF (YMAX.GE.B) PER = PER * (B-YMIN)/DY
  200 IF (IIN.EQ.1) PER=1.-PER
      NNN = IIX+(IIY-1)*NPTS
C (
      IF(INTCK.EQ.0) GO TO 201
      INTCK=0
      AINT=REAL(CU(NNN))**2 + AIMAG(CU(NNN))**2
      AINTMX=AMAX1(AINT,AINTMX)
      IF (AINT.NE.AINTMX) GO TO 201
      XINTMX=X
      YINTMX=Y
```

```
C )
  201 CU(NNN) = CU(NNN) * SQRT(PER)
  260 IF (XDISK.EQ.0..OR.IIN.EQ.1) GO TO 300
      IIN=1
      A = XDISK
      B = YDISK
      GO TO 199
C (
  300 FXF=1.
      IF(NREG.EQ.1.OR.NREG.EQ.2) FXF=1./WNOW**2
      AINTMX=AINTMX*FXF

      WRITE(6,310) AINTMX,XINTMX,YINTMX
  310 FORMAT(* THE MAX INTENSITY ON APERTURE PLATE IS IMAX= *,G13.5/
     1* AND IS LOCATED AT X= *,F13.5,*, Y= *,F13.5)
      RETURN
C ))
      END
```

6. SUBROUTINE BLUMIT

a. Purpose -- In the interstage duct, phase perturbation can be induced in the beam due to transient thermal blooming. This effect is suppressed by a sonic purge flow using the transverse thermal blooming routine. The BLUMIT routine models this residual sonic purge flow thermal blooming in the interstage duct. Figure 18 shows the subroutine BLUMIT organization.

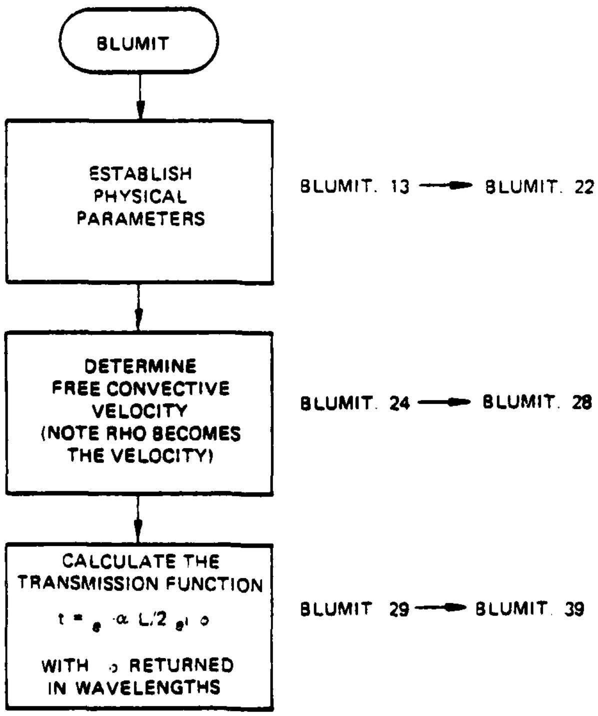


Figure 18. Subroutine BLUMIT organization.

b. Formalism -- As the beam propagates through the sonic purge flow, it is continuously distorted by that flow. Under the assumption that this distortion has a perturbative effect on the beam, the integrated effect of any thermal blooming can be approximated by a finite number of discrete steps in the following manner:

Assume each step is of length DL. The distortion is applied by propagating a length DL/2 to the center of the cell, then applying the thermal blooming transmission function. The beam is then propagated through the remaining DL/2 to the edge of the cell. The nonlinear blooming transmission function $t(x,y,\Delta L,I(x,y))$ is

$$t\left(x,y,\Delta L,I(x,y)\right) = \varepsilon^{-\alpha\Delta L/2}\,\varepsilon^{i\Delta\phi} \tag{31}$$

where, α is the absorptivity of the medium. $\Delta\phi$ is written

$$\Delta\phi = \frac{2\pi}{\lambda}\frac{dn}{dT}\int_0^{\Delta L} dz'\ \delta T\ (x,y,z') \tag{32}$$

This can be rewritten using the equation of state for an ideal gas ($P = RT\rho/M$) and the Gladstone-Dale relationship. Assuming constant pressure, the expression of $\Delta\phi$ becomes

$$\Delta\phi = \frac{2\pi}{\lambda}\left(-\frac{\rho C_{G-D}}{T}\right)\int_0^{\Delta L} dz'\ \delta T\ (x,y,z') \tag{33}$$

where δT represents the temperature variation in the flow due to heating by the beam. For transverse blooming, δT can be written

$$\delta T = \frac{\alpha}{\rho C p \nu_T}\int_{-\infty}^{x} dx'\ \ I\ (x',y,z) \tag{34}$$

In the above expression, the flow is assumed to be from the negative X direction with speed ν_T.

This effect is activated in subroutine CAVITY by setting NGTYPE=2. The duct is then treated as if it were another cavity, the gain/phase transmission function being that of transverse thermal blooming. It is updated by subroutine REGAIN.

Since the only mathematical difference between transverse and free convective is in the velocity, this routine can also handle free convection blooming with

$$v_{fc} = \left(\frac{2\alpha P(z) g}{\rho C p T} \right)^{1/3} \tag{35}$$

c. Fortran

<u>Argument list</u>

P = Intensity array. It returns as the phase change in wavelengths due to blooming.

G = Gain array. Intensity loss due to blooming.

NCV = Cavity number

WL = Wavelength

Commons modified - None

Subroutines called - None

The subroutine BLUMIT computer printout follows.

SUBROUTINE BLUMIT 76/176 OPT=1 FIN 4.6+452 04/27/79 12.23.47

```
      SUBROUTINE BLUMIT(P,G,NCV,WL)
C     INTERSTAGE DUCT THERMAL BLOOMING MODEL
C     THIS ROUTINE CALCULATES THE COMPLEX TRANSMISSION FUNCTION OF THE
C     ALL INTERSTAGE DUCT AS A FUNCTION OF THE POWER DENSITIES WITHIN
C     THE DUCT
C     THIS CODE IS PRELIMINARY
      LEVEL 2, P,G,XC,WL
      COMMON/CAV2/XC(5),YC(5),ZC(5),NX(5),NY(5),NS(5), XMC(5),YMC(5),
     1 NN(20),S2(196,5),TV1(5),TV2(5),TV3(5),TVN2(5),TSCAV(5),S3(35),
     2 TITLE(20),AVG(5),NSYM
      DIMENSION P( 1 ), G(1)
      ANGL = TSCAV(NCV)
      ALFA = TV1(NCV)
      CP = TV2(NCV)
      RHO = TV3(NCV)
      T = TVN2(NCV)
      DELZ = ZC(NCV)/NS(NCV)
```

```
      NXA = NX(NCV)
      NYA = NY(NCV)/(NSYM+1)
      MUT=NXA*NYA
      DELX = XC(NCV)/NXA
      IF(RHO.GT.1.) GO TO 10
      SUM = 0.
      DO 12 I=1,MUT
   12 SUM = SUM+P(I)
      SUM = SUM*DELX*YC(NCV)/NY(NCV)
      RHO =(980.665*SUM*ALFA/(RHO*CP*T))**(1./3.)
   10 CAP = .23*ALFA*DELZ*DELX/(CP*T*RHO)
      CAP2 = EXP (-ALFA *DELZ/2.)
      IB = +1
      IF(ANGL.GE.90.) IB=-1
      DO 20 J=1,NYA
      SUM = 0.
      DO 20 I=1,NXA
      IX = (1+NXA)*(1-IB)/2+IB*I + (J-1)*NXA
      SUM = SUM+ P(IX)
      P(IX)    = SUM*CAP
   20 G(I+(J-1)*NXA) = CAP2
      RETURN
      END
```

7. SUBROUTINE CAVITY

a. Purpose -- The CAVITY routine models the interaction of a GDL cavity and the complex optical field. As the simulated field is propagated through the cavity, it interacts with the flowing medium. As a result, both the intensity and phase of the beam are modified through the CAVITY routine. Figure 19 shows the subroutine CAVITY organization.

b. Formalism -- As the beam is propagated through the cavity, its intensity and phase are continuously updated. The beam's amplitude and phase are amplified and redirected by the medium-induced gain and phase change. This medium-beam interaction results in an integrated effect. It is assumed in CAVITY that the total effect can be approximated by a finite sum of N terms in the following manner: The total cavity length Z is divided into N steps, each Z/N = ΔL in length. In each segment, the interaction of the field with the medium is approximated by vacuum propagation through half of the segment, ($\Delta L/2$), followed by the application of a field dependent transmission function of the form

$$t(x,y,I) = \varepsilon^{\Delta L\left[\left(g(x,y,I)/2 + i\,\frac{2\pi}{\lambda}\,(\Delta n(x,y,I)\right)\right]} \qquad (36)$$

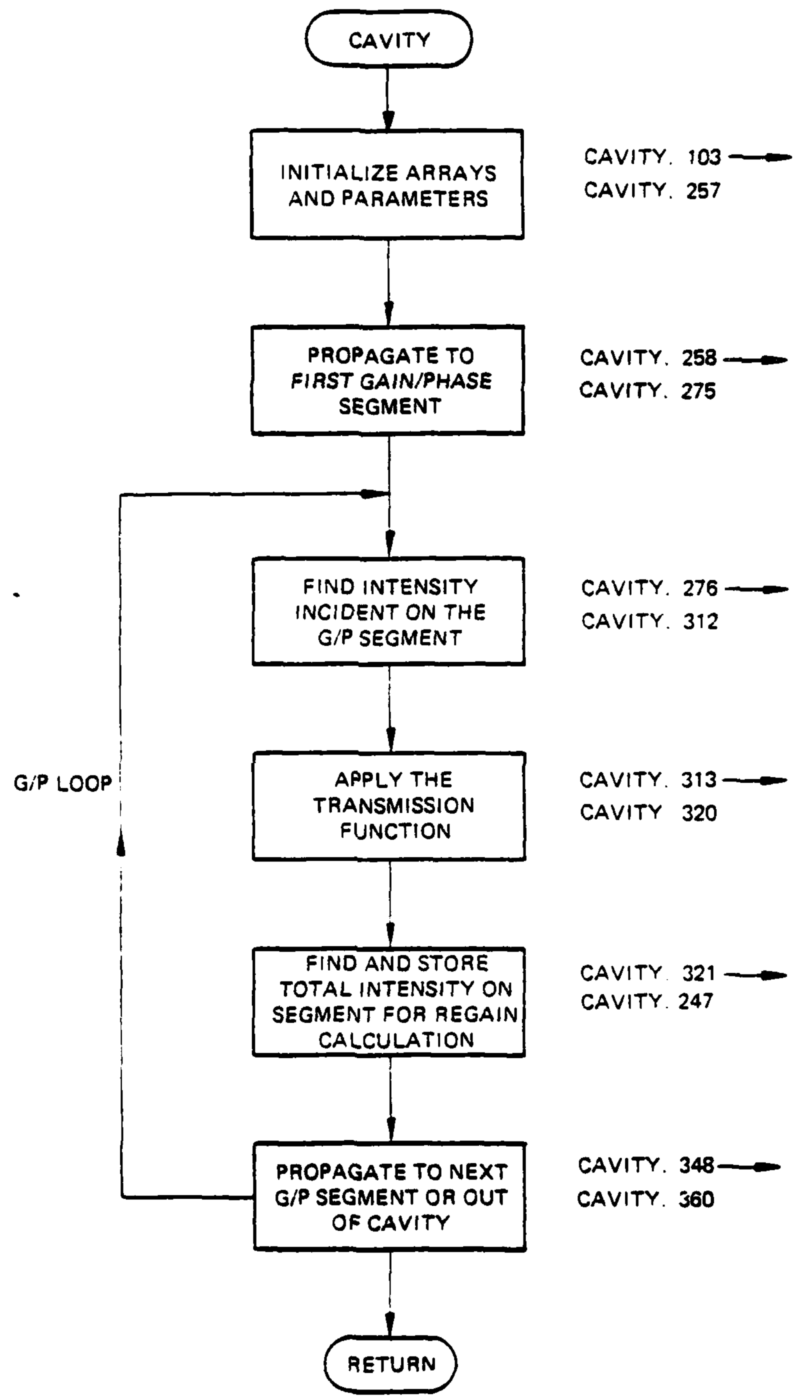


Figure 19. Subroutine CAVITY organization.

The gain coefficient g and refractive index Δn are calculated in other subroutines using an appropriate choice of kinetic modeling. The beam is then vacuum propagated through the remaining $\Delta L/2$. This procedure is repeated until the beam reaches the end of the cavity.

c. Fortran

Argument List

NCAV = Cavity identity number (1, 2, 3, ... N)

ILR = identifies the direction of propagation through the cavity:
- -1 => right to left
- +1 => left to right

NEWCAV = A parameter that identifies whether the cavity has been entered before.

INIT = .True. if it is the first interaction of a given run
= .False. if it is the second or subsequent interaction.

NSTE = Controlling parameter for subroutine STEP. If the geometric beam is converging or diverging, variable area mesh propagation (VAMP) should be used.

NSTE = 1 Constant mesh with setup
= 2 VAMP with setup (exit at end)
= 3 VAMP (setup and remain in VAMP)
= 4 VAMP (uses existing setup and exits)
= 5 VAMP (uses existing setup and does not exit)

IN = Input data set number or file from which data is to be read

RESTRT = .True. if initial beam is read in from unit IB
= .False. if analytical initial field is desired

NPLT = Controls plotting within cavity:
= 0 No plot
= 1 Print field before and after gain and gain coefficient

ZLI = Incoming propagation distance to cavity endwall
(Additional vacuum propagation distance)

ZLO = Exit propagation distance to cavity endwall
(Additional vacuum propagation distance)

Note: None of the parameters in the argument list is redefined by subroutine CAVITY.

Common variables altered:

- US - the intensity array
- PPD - interpolated power density
- CDUM - interpolated gain/phase transmission element
- XCAV - cavity coordinate array
- GFACT - define by namelist CAVTY2
- CFIL - redefined by its equivalence with Power Density array
- CU - the complex field - modified by propagation and the application of the cavity transmission function
- CG - defined for the first pass, read in for subsequent passes (Cavity gain/phase (G/P) array at each station within the cavity)

Namelist/CAVTY 2

CAVTY2 is used to initialize the cavity physical properties. The namelist is as follows:

```
      NAMELIST/ CAVTY2 /XLEN,YLEN,ZLEN,XMCAV,YMCAV,NODX,NODY,NOSEG,
     X FLAG,MREST, NGTYPE, NUPLOT, IUSE, IPUFN,T1,T2,T3,TN2,TS,PS,V,
     X PARCH,XN2,XCO2,XH2O,XCO,XO2,TITLE,ALFA,ACP,VELTY,TTEMP,ANGL,
     X AVGAIN, GFACTR
C
C      XLEN IS LENGTH OF CAVITY IN FLOW DIRECTION
C      YLEN IS LENGTH OF CAVITY ACROSS NOZZLES
C      ZLEN IS LENGTH OF CAVITY IN OPTICAL DIRECTION
C      XMCAV IS THE X-DIST OF OPTICAL AXIS FROM NOZZLE EXIT PLANE
C      YMCAV IS THE Y-DIST OF OPTICAL AXIS TO CAVITY AXIS
C      NODX  IS NUMBER OF GRID POINTS ALONG XLEN
C      NODY  IS NUMBER OF GRID POINTS ALONG YLEN
C      NOSEG IS NUMBER OF SEGMENTS, MAXIMUM OF 5 PER CAVITY
C      FLAG IS PARAMETER WHICH CONTROLS SELECTION OF DENSITY FIELD
C            = 1. SR-1,CONTOURED SIDEWALL
C            = 2. SR-1, FLAT SIDEWALL
C            = 3. ALL DENSITY
C            = 4. MOD-6. XLS-1
C            = 5. INPUT FROM CARDS ON DATA SET...IN...
C            = 6. SAME SPLINE CO-EFF THAT WERE READ IN FIVE
C          =8. RUN 112 AT T=1.6 SEC RIGHT STAGE BOTH WALLS
C         =8.1 READ NAMELIST DENS8 FOR RIGHT STAGE
C          =9. RUN 109 AT T=1.8 SEC LEFT STAGE BOTH WALLS
C         =9.1 READ NAMELIST DENS9 FOR LEFT STAGE
C         =10. READ DENSITY FIELD FROM UNIT 30
C         =11. READ DENSITY FIELD FROM UNIT 31
C      MREST  IS A FLAG FOR COMPUTING A RESTRATED GAIN...IF
C            = 1 READ OFF THE BIG G BUT USE NEW DENSTY FIELD
C            = 0 THEN TAKE THE CO-EFF AS THEY NOW EXIST
C
C      NGTYPE = 2...THERMAL BLOOMING FOR MULTI-BEAM
C             = 1...FULL BLOWN KINETICS...GDL
C             = 0   SIMPLE CLOSED FORM E.A.S. GDL KINETICS
```

```
C
C     NGPLOT = 0  NO PLOTS OF GAIN INSIDE THE CAVITY
C            = 1  PLOT A SLICE THROUGH THE CAVITY
C            = 2  ISO-AMPLITUDE OF GAIN IS PLOTTED
C            = 3  GET BOTH PLOTS
C            =-1  GET ALL POSSIBLE PLOTS
C
C       IPDEN = 0  NO PLOT OF POWER DENSITY AT EACH SLICE
C             = 1  SLICE PLOT OF PWR DENS
C             = 2  ISO- INTENSITY PLOT FOR CAVITY
C             = 3  ALL FOR THE MONEY
C     IUSE  = -1  NO FUSE NO PLOTS NO NOTHIN
C           = 0  NO FUSE ANAYLSIS,  BUT DENISTY GOULY PLOTS (AERO)
C           = 1  FUHS ANAYLSIS...NO PLOTS
C           = 2  FUHS IS USED (RHYME?) AS IS ISO-PLOTS
C           = 3  FUHS, ISO-PLOTS OF FUHS AND RESULTANT FUSE AND AERO
C        TITLE IS THE TILE TO APPEAR ON THE CAVITY GOULIES & GOULESESS
C
C     T1   IS VIBRATIONAL TEMPERTURE OF 00V       AT NEP, DEG K
C     T2   IS VIBRATIONAL TEMPERTURE OF 0V0       AT NEP, DEG K
C     T3   IS VIBRATIONAL TEMPERTURE OF V00       AT NEP, DEG K
C     TN2  IS VIBRATIONAL TEMPERTURE OF NITROGEN  AT NEP, DEG K
C     TS   IS STATIC TEMPERTURE IN CAVITY AT NEP, DEG K
C     PS   IS STATIC PRESSURE IN CAVITY AT NEP, ATM.
C     V    IS FLOW VELOCITY IN CAVITY AT NEP, CM/SEC
C     PBRCH IS P-BRANCH TRANSITION
C     XN2  IS MOLE FRACTION OF NITROGEN
C     XCO2 IS MOLE FRACTION OF CARBON DIOXIDE
C     XH2O IS MOLE FRACTION OF WATER
C     XCO  IS MOLE FRACTION OF CARBON MONOXIDE
C     XO2  IS MOLE FRACTION OF OXYGEN
C ****************  THERMAL BLOOMING MUTI-BEAM CAVITY      ***********
C       ALFA IS THE MEDIUM ABSORB CO-EFF IN CM-1
C       ACP IS THE MEDIUM SPECIFIC HEAT IN J/GM-DEG K
C     ITEMP IS THE MEDIUM TEMPERATURE IN DEG K
C     VELTY IS THE VELOCITY OF MEDIUM...IF .LT. 1. THEN THE FREE
C         CONVECTION VELOCOTY IS CALCULATED AND USED
C       ANGL IS THE ANGLE OF FLOW RELATIVE TO N.E.P.  0. IS LIKE CAVITY
C         (IF. AWAY FROM N.E.P.) AND 180. IS THE OTHER DIRECTION
C ***********************************************************************
C
C     AVGAIN IS THE AVERAGE OF GAIN CO-EFFIECENTS...HOOPE FAST CONVERGE
C
C
```

SUBROUTINE CAVITY   76/176   OPT=1   FIN 4.6+452   04/27/79   12.23.47

```
      SUBROUTINE CAVITY(NCAVN,ILR,NEWCAV,INIT,NSTE,IN,RESTRT,NPLT,       CAVITY     2
     X ZLI,ZLO)                                                          CAVITY     3
C     GDL CAVITY MODEL                                                   CAVITY     4
C     THIS ROUTINE APPLIES THE EFFECTS OF A GDL CAVITY TO THE COMPLEX    CAVITY     5
C     FIELD                                                              CAVITY     6
      LEVEL 2, CU,AC,XCAV,PDD,PPU,CUR,CG,US                              CAVITY     7
      LEVEL 2,PU                                                         CORR2      4
      COMMON /CUS/ US(17100)                                             CORR1     43
      COMMON /CAVX/ PDD,XCAV,CDUM                                        CORR1     44
      COMMON/MRPROP/RADCUR,ANGA,ANGY                                     CAVITY     9
      COMMON /GFACTR/ GFACT(2)                                           CROP1      9
      COMMON/WAY/WNOW,NREG,RAPTH                                         CAVITY    10
      COMMON/CAV2/  XC(5),YC(5),ZC(5),NX(5),NY(5),NS(5),XMC(5),YMC(5),   CAVITY    11
```

```
     2 NGTYP9(5),  NGPLO9(5),  IUS9(5),  IPDE9(5),                       CAVITY     12
     3           SSGAIN(190,5),SATIN(5),BETA(5),RMUS(5),                 CAVITY     13
     4 VEL(5),GAM(5),XMACH(5),TV1(5),TV2(5),TV3(5),TVN2(5),TSCAV(5),     CAVITY     14
     5 PSCAV(5),PH(5),FN2(5),FCO2(5),FH2O(5),FCO(5),FO2(5),TITLE(20),    CAVITY     15
     6 AVG(5),  NSYM                                                     CAVITY     16
      COMMON/MELT/CO(16384),CFIL(16512),X(128),WL,NPTS,NPY,DRX,DRY       CAVITY     17
      COMMON /CCG/ CG(17100)                                             CIODENS     1
      DIMENSION TPASS(290) , TPO(260) , PO(17100) , PPO(17100) ,         CIODENS     2
     X POD(2) , XCAV(190)   , COR(32768)                                 CIODENS     3
      COMPLEX CO,CFIL,CG,CARAY,CDUM                                      CAVITY     20
      LOGICAL INIT,RESTRT                                                CAVITY     21
      EQUIVALENCE (COR(1),CO(1))                                         CAVITY     22
      EQUIVALENCE (PPO(1),CO(1)) , (CFIL(1),PO(1))                       CIODENS     4
      DATA GFACTR / 1. /                                                 LROPI      10
      DATA  XLEN,YLEN,ZLEN,XMCAV,YMCAV,NODX,NODY,NOSEG,                  CAVITY     25
     X FLAG,MREST, NGTYPE, NGPLOT, IUSE, IPDEN,T1,T2,T3,TN2,TS,PS,V,     CAVITY     26
     X PBRCH,XN2,XCO2,XH2O,XCO,XO2,ALFA,ACP,VELTY,TTEMP,ANGL,AVGAIN      CAVITY     27
     X /5*0.0,2*0,3.0,0,3*0,-1,0,19*0.0/                                 CAVITY     28
C                                                                        CAVITY     29
      NAMELIST/ CAVTY2 /XLEN,YLEN,ZLEN,XMCAV,YMCAV,NODX,NODY,NOSEG,      CAVITY     30
     X FLAG,MREST, NGTYPE, NGPLOT, IUSE, IPDEN,T1,T2,T3,TN2,TS,PS,V,     CAVITY     31
     X PBRCH,XN2,XCO2,XH2O,XCO,XO2,TITLE,ALFA,ACP,VELTY,TTEMP,ANGL,      CAVITY     32
     X AVGAIN, GFACTR                                                    LROPI      11
C                                                                        CAVITY     34
C      XLEN IS LENGTH OF CAVITY IN FLOW DIRECTION                        CAVITY     35
C      YLEN IS LENGTH OF CAVITY ACROSS NOZZLES                           CAVITY     36
C      ZLEN IS LENGTH OF CAVITY IN OPTICAL DIRECTION                     CAVITY     37
C      XMCAV IS THE X-DIST OF OPTICAL AXIS FROM NOZZLE EXIT PLANE        CAVITY     38
C      YMCAV IS THE Y-DIST OF OPTICAL AXIS TO CAVITY AXIS                CAVITY     39
C      NODX  IS NUMBER OF GRID POINTS ALONG XLEN                         CAVITY     40
C      NODY  IS NUMBER OF GRID POINTS ALONG YLEN                         CAVITY     41
C      NOSEG IS NUMBER OF SEGMENTS, MAXIMUM OF 5 PER CAVITY              CAVITY     42
C      FLAG IS PARAMETER WHICH CONTROLS SELECTION OF DENSITY FIELD       CAVITY     43
C           = 1. SR=1,CONTOURED SIDEWALL                                 CAVITY     44
C           = 2. SR=1, FLAT SIDEWALL                                     CAVITY     45
C           = 3. ALL DENSITY                                             CAVITY     46
C           = 4. MOD=6, XLS=1                                            CAVITY     47
C           = 5. INPUT FROM CARDS ON DATA SET...IN...                    CAVITY     48
C           = 6. SAME SPLINE CO-EFF THAT WERE READ IN FIVE               CAVITY     49
C         =8. RUN 112 AT T=1.6 SEC RIGHT STAGE BOTH WALLS                SOQ77CY1    3
C        =8.1 READ NAMELIST DENS8 FOR RIGHT STAGE                        SOQ77CY1    4
C         =9. RUN 109 AT T=1.8 SEC LEFT STAGE BOTH WALLS                 SOQ77CY1    5
C        =9.1 READ NAMELIST DENS9 FOR LEFT STAGE                         SOQ77CY1    6
C        =10. READ DENSITY FIELD FROM UNIT 30                            SOQ77CY1    7
C        =11. READ DENSITY FIELD FROM UNIT 31                            SOQ77CY1    8
C      MREST  IS A FLAG FOR COMPUTING A RESTRATED GAIN...IF              CAVITY     50
C           = 1 READ OFF THE BIG G BUT USE NEW DENSTY FIELD              CAVITY     51
C           = 0 THEN TAKE THE CO-EFF AS THEY NOW EXIST                   CAVITY     52
C                                                                        CAVITY     53
C      NGTYPE = 2...THERMAL BLOOMING FOR MULTI-BEAM                      CAVITY     54
C             = 1...FULL BLOWN KINETICS...GDL                            CAVITY     55
C             = 0   SIMPLE CLOSED FORM E.A.S. GDL KINETICS               CAVITY     56
C                                                                        CAVITY     57
C      NGPLOT = 0  NO PLOTS OF GAIN INSIDE THE CAVITY                    CAVITY     58
C             = 1  PLOT A SLICE THROUGH THE CAVITY                       CAVITY     59
C             = 2  ISO-AMPLITUDE OF GAIN IS PLOTTED                      CAVITY     60
C             = 3  GET BOTH PLOTS                                        CAVITY     61
C             =-1  GET ALL POSSIBLE PLOTS                                CAVITY     62
C                                                                        CAVITY     63
C       IPDEN = 0  NO PLOT OF POWER DENSITY AT EACH SLICE                CAVITY     64
C             = 1  SLICE PLOT OF PWR DENS                                CAVITY     65
C             = 2  ISO- INTENSITY PLOT FOR CAVITY                        CAVITY     66
C             = 3  ALL FOR THE MONEY                                     CAVITY     67
C      IUSE  = -1  NO FUSE NO PLOTS NO NOTHIN                            CAVITY     68
C            = 0  NO FUSE ANAYLSIS,  BUT DENISTY GOULY PLOTS (AERO)      CAVITY     69
C            = 1  FUMS ANAYLSIS...NO PLOTS                               CAVITY     70
C            = 2  FUMS IS USED (RHYME?) AS IS ISO-PLOTS                  CAVITY     71
C            = 3  FUMS, ISO-PLOTS OF FUMS AND RESULTANT FUSE AND AERO    CAVITY     72
C        TITLE IS THE TILE TO APPEAR ON THE CAVITY GOULIES & GOULESESS   CAVITY     73
C                                                                        CAVITY     74
C      T1  IS VIBRATIONAL TEMPERTURE OF 00V        AT NEP, DEG K         CAVITY     75
C      T2  IS VIBRATIONAL TEMPERTURE OF 0V0        AT NEP, DEG K         CAVITY     76
```

```
C      T3  IS VIBRATIONAL TEMPERTURE OF V00        AT NEP, DEG K
C      TN2 IS VIBRATIONAL TEMPERTURE OF NITROGEN  AT NEP, DEG K
C      TS   IS STATIC TEMPERTURE IN CAVITY AT NEP, DEG K
C      PS   IS STATIC PRESSURE IN CAVITY AT NEP, ATM.
C      V    IS FLOW VELOCITY IN CAVITY AT NEP, CM/SEC
C      PBRCH IS P-BRANCH TRANSITION
C      XN2  IS MOLE FRACTION OF NITROGEN
C      XCO2 IS MOLE FRACTION OF CARBON DIOXIDE
C      XH2O IS MOLE FRACTION OF WATER
C      XCO  IS MOLE FRACTION OF CARBON MONOXIDE
C      XO2  IS MOLE FRACTION OF OXYGEN
C ****************     THERMAL BLOOMING MUTI-BEAM CAVITY     ************
C       ALFA IS THE MEDIUM ABSORB CO-EFF IN CM-1
C        ACP IS THE MEDIUM SPECIFIC HEAT IN J/GM-DEG K
C      TTEMP IS THE MEDIUM TEMPERATURE IN DEG K
C      VELTY IS THE VELOCITY OF MEDIUM...IF .LT. 1. THEN THE FREE
C          CONVECTION VELOCUTY IS CALCULATED AND USED
C       ANGL IS THE ANGLE OF FLOW RELATIVE TO N.E.P.  0. IS LIKE CAVITY
C          (IE. AWAY FROM N.E.P.) AND 180. IS THE OTHER DIRECTION
C ********************************************************************
C
C      AVGAIN IS THE AVERAGE OF GAIN CO-EFFIECENTS...HOOPE FAST CONVERGE
C
C
C
C *** TEST TO SEE IF BEEN IN THIS CAVITY BEFORE
       IF(.NOT.INIT.OR.NE*CAV.EQ.0) GO TO 50
       PI = 3.141592
       NSYM = 0
       IF (NPY.NE.NPTS) NSYM=1
       NNSYM=NSYM
       NOB= NPTS*NPY
       MREST  = 0
       READ(IN,CAVTY2)
       READ (IN,1243) TITLE
 1243 FORMAT (20A4)
       WRITE(6,200)
  200 FORMAT(39H0     ***********************************,/
      X          39H0               CAVITY PROPERTIES          ,/
      X          39H0     *********************************** )
       WRITE(6,100)  TITLE,XLEN,YLEN,ZLEN,NODX,NODY,NOSEG
  100 FORMAT(21H0CAVITY GEOMETRY FOR ,20A4/1X,7HXLEN = ,G12.5,4X,7HYLEN
      X= ,G12.5,4X,7HZLEN = ,G12.5,4X,6HNODX =,I6,4X,7HNODY = ,I5,4X,
      X8HNOSEG = ,I2)
       WRITE(6,101) XMCAV,YMCAV
  101 FORMAT(25H0LOCATION OF OPTICAL AXIS,/1X,8HXMCAV = ,G12.5,4X,
      X 8HYMCAV = ,G12.5)
       IF (NGTYPE.EQ.2) GO TO 106
       WRITE(6,102) TS,PS,V,PBRCH
  102 FORMAT(18H0CAVITY CONDITIONS,/1X,5HTS = ,G12.5,4X,5HPS = ,G12.5,
      X4X,11HVELOCITY = ,G12.5,4X,9HP-BRANCH ,F3.0)
       WRITE(6,103) XN2,XCO2,XH2O,XCO,XO2
  103 FORMAT(12H0COMPOSITION,/1X,6HXN2 = ,G12.5,4X,7HXCO2 = ,G12.5,4X,
      X7HXH2O = ,G12.5,4X,6HXCO = ,G12.5,4X,6HXO2 = ,G12.5)
C      LOAD CAVITY PARAMETERS INTO APPROPRIATE STORAGE ARRAYS
       TV1(NCAVN)=T1
       TV2(NCAVN)=T2
       TV3(NCAVN)=T3
       TVN2(NCAVN)=TN2
       TSCAV(NCAVN)=TS
       WRITE(6,104) TN2,T1,T2,T3
  104 FORMAT(25H0VIBRATIONAL TEMPERATURES,/1X,6HTN2 = ,G12.5,4X,5HT1 = ,
      XG12.5,4X,5HT2 = ,G12.5,4X,5HT3 = ,G12.5)
       GO TO 107
  106 MREST = 2
       TV1(NCAVN)=ALFA
       TV2(NCAVN)=ACP
       TV3(NCAVN)=VELTY
       TVN2(NCAVN)=TTEMP
       TSCAV(NCAVN)=ANGL
       WRITE(6,133) ALFA,ACP,VELTY,TTEMP,ANGL
```

```
  193 FORMAT (//67H   THERMAL BLOOMING ANAYLSIS OF MULTI-BEAM SYSTEM...C CAVITY     147
     XONSTANTS ARE :,/7H ALFA =,G12.5,6H  CP =,F8.4,17H  FLOW VELOCITY = CAVITY     148
     X,F8.4,       9H  TEMP = ,F8.4,      10H  ANGLE = ,F8.4 //  )            CAVITY     149
  107 YC(NCAVN) = YLEN                                                        CAVITY     150
      XC(NCAVN) = XLEN                                                        CAVITY     151
      ZC(NCAVN) = ZLEN                                                        CAVITY     152
      AVG(NCAVN)=AVGAIN                                                       CAVITY     153
      XMC(NCAVN)=XMCAV                                                        CAVITY     154
      YMC(NCAVN)=YMCAV                                                        CAVITY     155
      NX(NCAVN) = NODX                                                        CAVITY     156
      NY(NCAVN) = NODY                                                        CAVITY     157
      NS(NCAVN) = NOSEG                                                       CAVITY     158
      GFACT(NCAVN)=GFACTN                                                     CHOP1       12
      DCZ = ZC(NCAVN)/NS(NCAVN)                                               CAVITY     159
      DCX = XC(NCAVN)/NX(NCAVN)                                               CAVITY     160
      DCY = YC(NCAVN)/NY(NCAVN)                                               CAVITY     161
      NSA=NS(NCAVN)                                                           CAVITY     162
      NYA=NY(NCAVN)/(NSYM+1)                                                  CAVITY     163
      NXA=NX(NCAVN)                                                           CAVITY     164
      MUT=NXA*NYA                                                             CAVITY     165
      NGTYP9(NCAVN)=NGTYPE                                                    CAVITY     166
      NGPLO9(NCAVN)=NGPLOT                                                    CAVITY     167
      IUS9(NCAVN)=IUSE                                                        CAVITY     168
      IPDE9(NCAVN)=IPDEN                                                      CAVITY     169
      PSCAV(NCAVN)=PS                                                         CAVITY     170
      VEL(NCAVN)=V                                                            CAVITY     171
      PB(NCAVN)=PBRCH                                                         CAVITY     172
      CARAY = CMPLX(0.,2.*PI/WL)                                              CAVITY     173
      TOPIWL = 2. * PI / WL                                                   SOU77CY1     9
      FN2(NCAVN)=XN2                                                          CAVITY     174
      FCO2(NCAVN)=XCO2                                                        CAVITY     175
      FH2O(NCAVN)=XH2O                                                        CAVITY     176
      FCO(NCAVN)=XCO                                                          CAVITY     177
      FO2(NCAVN)=XO2                                                          CAVITY     178
      IBASE = 10+(NCAVN-1)*11                                                 CAVITY     179
      IF (NGTYPE.EQ.2) GO TO 108                                              CAVITY     180
C     CALCULATE SMALL SIGNAL GAIN AS A FUNCTION OF X                          CAVITY     181
      CALL GAINXY(PO,US,NCAVN,1)                                              CAVITY     182
      WRITE (7)  (CO(IZ),IZ=1,NOB)                                            CAVITY     183
      REWIND 7                                                                CAVITY     184
      RHO=RHOS(NCAVN)                                                         CAVITY     185
C     CALCULATE CAVITY DENSITY FIELD AS A FUNCTION OF X AND Y                 CAVITY     186
      CALL DENSY(FLAG,RHO,XLEN,YLEN,DCZ,NXA,NYA,1,IN,NNSYM)                   CORR1       47
C     STORE DENSITY FIELD ON DIRECT ACCESS FILE                               CAVITY     188
      WRITE(IBASE)  (PPD(IZ),IZ=1,MUT)                                        CAVITY     189
      REWIND IBASE                                                            CAVITY     190
C                                                                             CAVITY     191
C   IF RESTARTING FROM A PREVIOUS RUN, THEN SKIP THE INITIAL                  CAVITY     192
C   GUESS AT GAIN                                                             CAVITY     193
C                                                                             CAVITY     194
  108 IF ( RESTRT .AND. MREST  .NE. 1) GO TO 49                               CAVITY     195
      DO 10 NNS=1,NSA                                                         CAVITY     196
      XCLO=-DCX/2.                                                            CAVITY     197
      IBASE = IBASE+1                                                         CAVITY     198
      IF (MREST .NE.1) GO TO 20                                               CAVITY     199
      READ(IBASE)  (CG(IZ),IZ=1,MUT)                                          CAVITY     200
      REWIND IBASE                                                            CAVITY     201
C     GENERATE COMPLEX GAIN ARRAYS                                            CAVITY     202
   20 XMUL1  = DCZ/6.                                                         CAVITY     203
      DO 11 IX=1,NXA                                                          CAVITY     204
      XCLO= DCX+XCLO                                                          CAVITY     205
      GOP = SSGAIN(IX,NCAVN)                                                  CAVITY     206
      XMULTR = EXP (XMUL1*GOP)                                                CAVITY     207
      DO 11 IY=1,NYA                                                          CAVITY     208
      IZ = IX+(IY-1)*NXA                                                      CAVITY     209
      PHIM = TOPIWL * PPD(IZ)                                                 SOU77CY1    10
      IF (MREST .EQ.0)                                                        CAVITY     210
     XCG(IZ) = XMULTR*CMPLX(COS(PHIM),SIN(PHIM))                              SOU77CY1    11
C    X  CG( IZ  ) = EXP(GOP*DCZ/6.)*CEXP(CARAY*PPD( IZ  ))                    CAVITY     212
      IF (MREST .EQ.1)                                                        CAVITY     213
     X CG(IZ) = CABS(CG(IZ))*CMPLX(COS(PHIM),SIN(PHIM))                       SOU77CY1    12
C    X  CG( IZ  ) = CABS(CG( IZ  ))*CEXP(CARAY*PPD( IZ  ))                    CAVITY     215
```

```
      IF (MREST .EQ.2)
     X     CG( IZ  ) = CMPLX(1.,0.)
   11 CONTINUE                                                          CAVITY
      WRITE(IBASE) (CG(IZ),IZ=1,MUT)                                    CAVITY
   10 REWIND IBASE                                                      CAVITY
   49 READ (7) (CU(IZ),IZ=1,NOH)                                        CAVITY
      REWIND 7                                                          CAVITY
C     APPLICATION OF CAVITY TRANSMISSION FUNCTIONS TO COMPLEX FIELD     CAVITY
   50 NSA=NS(NCAVN)                                                     CAVITY
      NYA=NY(NCAVN)/(NSYM+1)                                            CAVITY
      NXA=NX(NCAVN)                                                     CAVITY
      MUT = NXA*NYA                                                     CAVITY
C *** FIRST TIME THROUGH THIS CAVITY, ZERO AVERAGE INTENSITY ARRAY      CAVITY
      IF(NEWCAV.EQ.0) GO TO 51                                          CAVITY
C     CALL ZERO(PO( 1 ),PO(  MUT ))                                     CAVITY
      DO 485 IZERO=1,MUT                                                CAVITY
  485 PO(IZERO)=0.                                                      CAVITY
      IBASE=10+(NCAVN-1)*11+5                                           CAVITY
      NCOLD = 0                                                         CAVITY
      DO 53 IZ=1,NSA                                                    CAVITY
      IBAS=IBASE+IZ                                                     CAVITY
      WRITE (IBAS) (PO(IZ),IZ=1,MUT)                                    CAVITY
   53 REWIND IBAS                                                       CAVITY
   51 IBASE = 10+(NCAVN-1)*11                                           CAVITY
      IF (NCAVN .EQ. NCOLD) GO TO 26                                    CAVITY
      DX = XC(NCAVN)/NXA                                                CAVITY
      DY = YC(NCAVN)/NY(NCAVN)                                          CAVITY
C     ESTABLISH CAVITY INTERPOLATION ARRAY (TPASS)                      CAVITY
      TPASS(1) = DX                                                     CAVITY
      TPASS(2) = DY                                                     CAVITY
      TPASS(3) = NYA+.001                                               CAVITY
      TPASS(4) = NXA+.001                                               CAVITY
      TPASS(5) = (DY-YC(NCAVN))/2. + YMC(NCAVN)                         CAVITY
      TPASS(5+NYA) =      DX/2. - XMC(NCAVN)                            CAVITY
      DO 5 I = 2,NYA                                                    CAVITY
    5 TPASS(4+I) = TPASS(3+I) + DY                                      CAVITY
      DO 6 N = 2,NXA                                                    CAVITY
    6 TPASS(4+NYA+N) =TPASS(3+NYA+N)+  DX                               CAVITY
      NCOLD = NCAVN                                                     CAVITY
   26 NST=NSTE                                                          CAVITY
      IOUT=1                                                            CAVITY
      DCZ = ZC(NCAVN)/NSA                                               CAVITY
C     PROPAGATE TO FIRST GAIN/PHASE SEGMENT                             CAVITY
      IF (NSTE.EQ.3.OR.NSTE.EQ.5) IOUT=0                                CAVITY
      IF (NSTE.EQ.3) NST=2                                              CAVITY
C     IF (NSTE.GE.4.AND.(DCZ/2.+ZLI).GT.1.0) CALL CORE(DCZ/2.0+ZLI,0,0) CAVITY
      IF (NSTE.GE.4.AND.(DCZ/2.+ZLI).GT.1.0) CALL STEP(DCZ/2.0+ZLI,     CAVITY
     X RADCUR,.1,.1,NST,0,0,ANGX,ANGY,0,1)                              CAVITY
      IF (NSTE.LE.3.AND.(DCZ/2.+ZLI).GT.1.0)                            CAVITY
     1CALL STEP(DCZ/2.0+ZLI,RADCUR,.1,.1,NST, 0,0,ANGX,ANGY,0,0)        CAVITY
      MEMORY=0                                                          CAVITY
      IF (NSTE.LE.3.AND.(DCZ/2.+ZLI).LE.1.0)MEMORY=1                    CAVITY
      DO 55 JNS=1,NSA                                                   CAVITY
      IB = 0                                                            CAVITY
      IF(ILR.LT.0) IB=NS(NCAVN)+1                                       CAVITY
      IADD = JNS*ILR+IB                                                 CAVITY
      XFACT=1.                                                          CAVITY

      IF (NREG.NE.0) XFACT=1./WNOW**2                                   CAVITY
      IDPO = IADD+5+IBASE                                               CAVITY
C     ESTABLISH FIELD INTERPOLATION ARRAY                               CAVITY
      TPO(1) = X(2)-X(1)                                                CAVITY
      TPO(2) = TPO(1)                                                   CAVITY
      TPO(3)= NPY                                                       CAVITY
      TPO(4) = NPTS                                                     CAVITY
      DO 54 IPJ=1,NPY                                                   CAVITY
   54 TPO(IPJ+4) = X(IPJ)+ORY                                           CAVITY
      DO 82 IPJ=1,NPTS                                                  CAVITY
   82 TPO(IPJ+NPY+4 ) = X(IPJ)+ORX                                      CAVITY
C *** COMPUTE INTENSITY INCIDENT UPON SEGMENT                           CAVITY
      DO 61 MA=1,NOH                                                    CAVITY
```

```
   61 US( MX  ) = (CUR(2*MX-1)**2 + CUR(2*MX)**2) * XFACT                  CAVITY     286
      WRITE (7) (US(IZ),IZ=1,NOB)                                            CAVITY     287
      REWIND 7                                                               CAVITY     288
      IDCG = IADD+IBASE                                                      CAVITY     289
      READ(IDCG) (CG(IZ),IZ=1,MUT)                                           CAVITY     290
      REWIND IDCG                                                            CAVITY     291
      IF (NPLT.EQ.0) GO TO 68                                                CAVITY     292
C     PLOT FIELD INCIDENT ON GAIN/PHASE SEGMENT                              CAVITY     293
      WRITE (6,69) NCAVN,IADD                                                CAVITY     294
   69 FORMAT(39H1  *****     E-M FIELD IN CAVITY NUMBER ,I2,19H      AT S CAVITY     295
     XEGMENT # ,I2,41H   BEFORE GAIN HAS BEEN APPLIED     ***** ,/)          CAVITY     296
      K=1                                                                    CAVITY     297
      UMAX=0.0                                                               CAVITY     298
      CALL OUTPUT(CU,NPY,NPTS,X,K,UMAX,.TRUE.,.FALSE.,.FALSE.)               CAVITY     299
C     PLOT GAIN PROFILE THROUGH CENTER OF CAVITY                             CAVITY     300
      WRITE (6,67) NCAVN,IADD                                                CAVITY     301
   67 FORMAT(90H1  CG(I,J) PLOTTED IN THE X-DIRECTION THROUGH THE CENTER CAVITY     302
     X OF THE CAVITY,       FOR CAVITY # ,I2,15H     SEGMENT # ,I2)          CAVITY     303
      DELXC=XC(NCAVN)/NX(NCAVN)                                              CAVITY     304
      XCAV(1)=DELXC/2.                                                       CAVITY     305
      DO 667 KCX=2,NXA                                                       CAVITY     306
  667 XCAV(KCX)=XCAV(KCX-1)+DELXC                                            CAVITY     307
      K=-1                                                                   CAVITY     308
      UMAX=0.0                                                               CAVITY     309
      CALL OUTPUT(CG,NY(NCAVN),NX(NCAVN),XCAV,K,UMAX,.TRUE.,.FALSE.,        CAVITY     310
     X    .FALSE.)                                                           CAVITY     311
   68 IZ=0                                                                   CAVITY     312
C     APPLY CAVITY TRANSMISSION TO COMPLEX FIELD                             CAVITY     313
      DO 58 JY=1,NPY                                                         CAVITY     314
      DO 58 JX=1,NPTS                                                        CAVITY     315
      CALL INTERP(TPASS,X(JX)*DRX,X(JY)*DRY,CG,2,CDUM,NNSYM)                 CORRI       48
      IZ = IZ+1                                                              CAVITY     317
   58 CU( IZ  ) = CDUM*CU( IZ  )                                             CAVITY     318
      READ (7) (US(IZ),IZ=1,NOB)                                             CAVITY     319
      REWIND 7                                                               CAVITY     320
C     CALCULATE SUM OF INTENSITIES BEFORE AND AFTER GAIN/PHASE SEGMENT       CAVITY     321
      DO 64 JY=1,NOB                                                         CAVITY     322
   64 US(JY) =(CUR(2*JY-1)**2 + CUR(2*JY)**2)* XFACT+US(JY)                  CAVITY     323
      READ (IDPD) (PD(IZ),IZ=1,MUT)                                          CAVITY     324
      REWIND IDPD                                                            CAVITY     325
      IF (NPLT.EQ.0) GO TO 73                                                CAVITY     326
C     PLOT FIELD LEAVING GAIN/PHASE SEGMENT                                  CAVITY     327
      WRITE (6,39) NCAVN,IADD                                                CAVITY     328
   39 FORMAT(///36H1  ***** E-M FIELD IN CAVITY NUMBER ,I2,19H      AT S CAVITY     329

     XEGMENT # ,I2,40H    AFTER GAIN HAS BEEN APPLIED   ***** /)            CAVITY     330
      K=1                                                                    CAVITY     331
      UMAX=0.0                                                               CAVITY     332
      CALL OUTPUT(CU,NPY,NPTS,X,K,UMAX,.TRUE.,.FALSE.,.FALSE.)               CAVITY     333
   73 TPDMIN=TPD(5)                                                          CAVITY     334
C     INTERPOLATE POWER DENSITIES ONTO CAVITY GRID, SUM WITH RESULTS         CAVITY     335
C     OF PREVIOUS PASSES AND STORE                                           CAVITY     336
      DO 57 INY=1,NYA                                                        CAVITY     337
      TTESY=TPASS(4+INY)                                                     CAVITY     338
      IF (TTESY.LT.TPDMIN)  GO TO 57                                         CAVITY     339
      DO 56 INX=1,NXA                                                        CAVITY     340
      TTESX=TPASS(4+NYA+INX)                                                 CAVITY     341
      CALL INTERP(TPD,TTESX,TTESY,US,1,PDD,NNSYM)                            CORRI       49
      IZ = INX+(INY-1)*NXA                                                   CAVITY     343
   56 PD(   IZ  ) = PD(  IZ   )+PDD(1)/2.                                    CAVITY     344
   57 CONTINUE                                                               CAVITY     345
      WRITE(IDPD) (PD(IZ),IZ=1,MUT)                                          CAVITY     346
      REWIND IDPD                                                            CAVITY     347
C     PROPAGATE TO NEXT GAIN/PHASE SEGMENT                                   CAVITY     348
C     IF(JNS.NE.NSA.AND.MEMORY.EQ.0)CALL CORE(DCZ,0,0)                       CAVITY     349
      IF(JNS.NE.NSA.AND.MEMORY.EQ.0)CALL STEP(DCZ,RADCUR,.1,.1,NST,0,      CAVITY     350
     X 0,ANGX,ANGY,0,1)                                                      CAVITY     351
      IF(JNS.NE.NSA.AND.MEMORY.EQ.1)                                         CAVITY     352
     1CALL STEP(DCZ,RADCUR,.1,.1,NST, 0,0,ANGX,ANGY,0,0)                     CAVITY     353
      MEMORY=0                                                               CAVITY     354
C     PROPAGATE OUT OF CAVITY                                                CAVITY     355
C     IF(JNS.EQ.NSA.AND.(DCZ/2.+ZLO).GT.1.0)CALL CORE(DCZ/2.0+ZLO,IOUT,0 CAVITY     356
```

```
C     X)                                                              CAVITY     357
       IF(JNS.EQ.NSA.AND.(DCZ/2.*ZLO).GT.1.0)CALL STEP(DCZ/2.0*ZLO,    CAVITY     358
     X RADCUR,.1,.1,NST,IOUT,0,ANGA,ANGY,0,1)                         CAVITY     359
   55 CONTINUE                                                        CAVITY     360
      RETURN                                                          CAVITY     361
      END                                                             CAVITY     362
```

8. SUBROUTINE CENBAR

a. Purpose -- This subroutine is used by QUAL to find the centroid coordinates of the far-field beam. Figure 20 describes subroutine CENBAR organization.

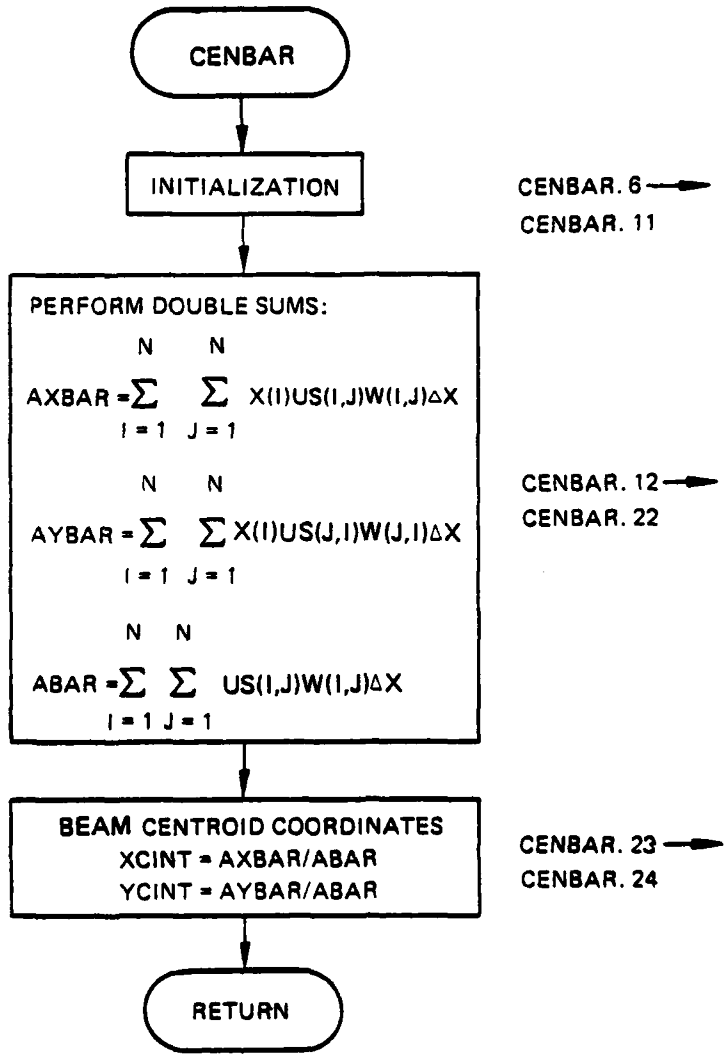


Figure 20. Subroutine CENBAR organization.

b. Formalism -- Let $E(x,y)$ represent the field and let $w(x,y)$ be a weighting function defined by

$$w(x,y) = \begin{cases} 1, \text{ if } |E(x,y)|^2 > 0.1 \; (|E|^2_{max}) \\ 0, \text{ if } |E(x,y)|^2 \leq 0.1 \; (|E|^2_{max}) \end{cases} \tag{37}$$

Then the intensity-weighted centroid coordinates are found from

$$\vec{x}_c = \frac{\iint dxdy \; |E(x,y)|^2 \; w(x,y) \; \vec{x}}{\iint dxdy \; |E(x,y)|^2 \; w(x,y)} \tag{38}$$

where the integrals are numerically evaluated over the calculation region.

c. Fortran

Argument List

NPTS = Number of points in x direction

DX = spacing between two adjacent points

X = coordinate array

US = intensity array = $|CU(I)|^2 = |E(x,y)|^2$

$$\left.\begin{array}{l} \text{XCINT = Centroid coordinate in the X direction} \\ \text{YCINT = Centroid coordinate in the Y direction} \end{array}\right\} \vec{x}_c$$

UMAX = Maximum Intensity

The incoming parameters are NPTS,DX,X,US,UMAX. They are unchanged by this routine and are used to calculate XCINT and YCINT.

Note: The subroutine assumes that the field is square. Computer printout of subroutine CENBAR follows.

SUBROUTINE CENBAR 76/175 OPT=1 FIN 4.6+452 04/27/79 12.23.47

```
      SUBROUTINE CENBAR ( NPTS, DX, X, US, XCINT, YCINT, UMAX)          CENBAR
C     CENTROID LOCATION MODEL                                           CENBAR
C     THIS ROUTINE LOCATES THE INTENSITY WEIGHTED CENTROID OF THE       CENBAR
C     COMPLEX FIELD                                                     CENBAR
      LEVEL 2, NPTS,X,US                                                CENBAR
      DIMENSION X(1), US(1)                                             CENBAR
      AXBAR=0.                                                          CENBAR
      UCUT = .1 * UMAX                                                  CENBAR
      AYBAR=0.                                                          CENBAR
```

```
      ABAR=0.                                                    CENBAR
      DO 10 I=1,NPTS                                             CENBAR
      ADY=0.                                                     CENBAR
      ADX=0.                                                     CENBAR
      DO 11 J=1,NPTS                                             CENBAR
      IJ = I +(J-1)*NPTS                                         CENBAR
      JI = J+(I-1)*NPTS                                          CENBAR
      IF (US(IJ ) .GT. UCUT )  ADX = ADX + US(IJ )               CENBAR
   11 IF (US(JI ) .GT. UCUT )  ADY = ADY + US(JI )               CENBAR
      AXBAR=AXBAR+ADX*DX*X(I)                                    CENBAR
      AYBAR=AYBAR+ADY*DX*X(I)                                    CENBAR
   10 ABAR=ABAR+ADX*DX                                           CENBAR
      XCINT=AXBAR/ABAR                                           CENBAR
      YCINT=AYBAR/ABAR                                           CENBAR
      RETURN                                                     CENBAR
      END                                                        CENBAR
```

9. SUBROUTINE DENSY

Called from: CAVITY

Calls: LINTERP, ROSN, ROSN6

a. Purpose -- This routine controls the generation of the cavity density-induced phase distortion for each cavity in the optical train. DENSY provides a choice of density fields including interpreted test data from several devices and the ability to read in density fields from tape. Little formal calculation is done within the routine itself, other than the generation of multipliers and certain other constants used by the interpolation routines. DENSY does tabulate spline coefficients if any are used to generate the phase distorting field, and provides a decile plot of the phase field. Figure 21 shows the subroutine DENSY flow chart.

Argument List

FLAG	flag for density field selection
IF	file number where MOD 6 density field is stored
IN	file number where input card data is stored
NPX	number of cavity density grid points in X direction
NPY	number of cavity density grid points in Y direction
NSYM	flag for symmetry of field
RHO	free stream static density
XLEN	X-dimension (flow direction) of cavity segment
YLEN	Y-dimension (sidewall-to-sidewall) of cavity segment
ZSLAB	Z-dimension (optical direction) of cavity segment

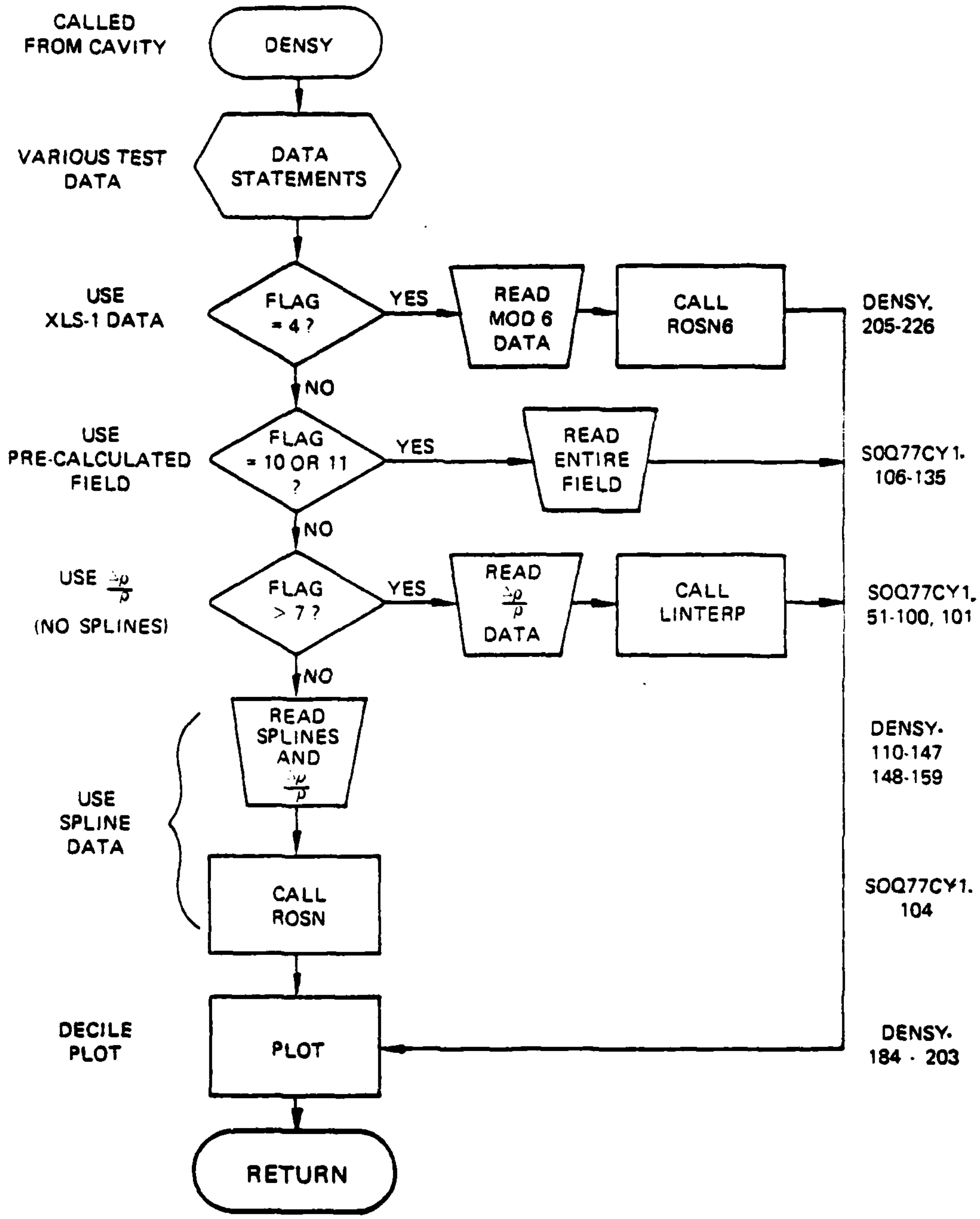


Figure 21. Subroutine DENSY flow chart.

Commons Modified

/MELT/

Variables Modified

P	storage array for density induced phase distribution
X4, Y4, Z4, C4, M4, N4, ROCL	spline coefficient and other data useful in generation of MOD 6 (XLS-1) density field - not used for other field options

/LENSY/

Variables Modified

D	spline coefficient array
H	cavity width (sidewall-to-sidewall)
LL	flag for cavity wall symmetry
M	number of data points in spline arrays
RHOCL	centerline density variation
TITLE	field identified
TM	tangent of Mach angle
XLS	spline array center deviation from NEP
XMULT	magnifier for entire density field
Y	position array
Z	density change array

b. **Relevant formalism** -- Most of the formal calculations involving spline fitting a density field and interpolating the results are done external to DENSY (see subroutines LINTERP, ROSN, and ROSN6). This routine directs the activities that generate the desired field. These activities are summarized below:

(1) The density field is read in directly from information generated by another program and written to disk (FLAG = 10 or 11)

(2) The sidewall density variations, but not the coefficients for a spline fit, are read in by NAMELIST or from data statements. The complete density field is generated by projecting these data into the flow along Mach lines, and linearly interpolating via LINTERP. (FLAG = 8, 8.1, 9, 9.1)

(3) The sidewall density variations and their spline fit coefficients are read in on cards or taken from DATA statements. The complete density field is generated by interpolating with the spline fit along the projection. (FLAG = 1 through 7)

A decile plot of the density-induced optical path variation (in cm) is generated after returning from one of these actions.

Subroutine DENSY computer printouts follow.

```
SUBROUTINE DENSY     76/176      OPT=1      FIN 4.6+452       04/27/79   12.23.47

      SUBROUTINE DENSY(FLAG,RHO,XLEN,YLEN,ZSLAB,NPX,NPY,IF,IN,NSYM)      DENSY        2
C THIS PROGRAM COMPUTES PHASE VARIATION IN EACH SEGMENT DUE TO          DENSY        3
C VARIATIONS IN THE GAS DENSITY IN THE OPTICAL CAVITY. INPUT PARAMETERS  DENSY        4
C ARE:                                                                  DENSY        5
C     RHO  - FREE STREAM STATIC DENSITY                                 DENSY        6
C     XLEN,YLEN,ZSLAB ARE DIMENSIONS OF SEGMENT                         DENSY        7
C     NPX,NPY ARE NUMBER OF GRID POINTS IN X,Y DIMENSIONS               DENSY        8
C     IF IS THE FILE ON WHICH THE MOD 6 DENSITY FIELD IS STORED         DENSY        9
C     FLAG - FLAG FOR DENSITY FIELD SELECTION                           DENSY       10
C          =  1. FOR CONTOURED SIDEWALL,T=3 SEC                         DENSY       11
C          =  2. FOR FLAT SIDEWALL,T=3 SEC                              DENSY       12
C          =  3. LATEST AND GREATEST TWO STAGE DENSITY FIELD            DENSY       13
C          = 4. FOR XLS-1 MOD 6 NOZZLES  NORTH  AND SOUTH SIDE          DENSY       14
C          = 5. FOR INPUT FROM CARDS OF SPLINE CO-EFFS.                 DENSY       15
C          = 6. FOR INPUT FROM READ IN IN PREVIOUS CAVITY DEFINITION    DENSY       16
C       =8. RUN 112 AT T=1.6 SEC RIGHT STAGE BOTH WALLS                 SOU77CY1    13
C      =8.1 READ NAMELIST DENS8 FOR RIGHT STAGE                         SOU77CY1    14
C       =9. RUN 109 AT T=1.8 SEC LEFT STAGE BOTH WALLS                  SOU77CY1    15
C      =9.1 READ NAMELIST DENS9 FOR LEFT STAGE                          SOU77CY1    16
C      =10. READ DENSITY FIELD FROM UNIT 30                             SOU77CY1    17
C      =11. READ DENSITY FIELD FROM UNIT 31                             SOU77CY1    18
C                                                                       DENSY       17
C                                                                       DENSY       18
      IMPLICIT COMPLEX(C)                                               DENSY       19
      LEVEL 2, P                                                        DENSY       20
      REAL C4                                                           DENSY       21
      EQUIVALENCE (MA,M)                                                DENSY       22
      COMMON /MELT/ P(16384),                                           DENSY       23
     X        X4(21),Y4(21,81),Z4(21,81),C4(21,81),M4(21),N4,ROCL       DENSY       24
     X,DUMYS(44394)                                                     CORR2        5
      DIMENSION TITLE1(20),TITLE2(20),                                  DENSY       25
     X           Y1(50),Z1(50),D1(50),Y2(45),Z2(45),D2(45),             DENSY       26
     X           Y3(50),Z3(50),D3(50),TITLE3(20)                        DENSY       27
     X        ,TITLE8(20),Y8(50),Z8(50),D8(50)                          SOU77CY1    19
     X        ,Y8W(50),Z8W(50),D8W(50)                                  SOU77CY1    20
     X        ,Y9(50),Z9(50),D9(50)                                     SOU77CY1    21
     X        ,Y9W(50),Z9W(50),D9W(50),TITLE9(20)                       SOU77CY1    22
      DIMENSION TILE(12),IP(190)                                        CORR1       50
      COMMON/DENSY/Y(51,2),Z(51,2),D(51,2),TM(2),XLS(2),M,XMULT(2),     DENSY       29
```

```
X    RHOCL(2),M(2),TITLE(20),LL                                         DENSY      30
  NAMELIST /DENS8/ TM8,M8,XM8,M8W,Y8,Z8,Y8W,Z8W                         SOQ77CY1   23
  NAMELIST /DENS9/ TM9,M9,XM9,M9W,Y9,Z9,Y9W,Z9W                         SOQ77CY1   24
 DATA GDC /0.228/                                                       DENSY      31
 DATA TM3/.21034/, M3/50/, XM3/.01/                                     DENSY      32
 DATA Y1/-5.,-4.,-3.,-2.3,-2.2,-2.1,-2.05,-2.,-1.95,-1.90,-1.85,        DENSY      33
A-1.8,-1.7,-1.35,-1.,-.9,-.8,-.7,-.65,-.6,-.5,-.45,-.4,-.35,-.30,       DENSY      34
B-.25,-.2,-.15,-.1,-.05,0.,.05,.1,.15,.2,.25,.3,.35,.4,.5,.6,.65,       DENSY      35
C.7,.75,.85,1.,2.,3.,4.,5./                                             DENSY      36
 DATA Z1/5*.004,.0035,.003,.002,.001,.0005,7*0.,.001,.002,.007,.014     DENSY      37
A,.017,.015,.006,-.006,-.014,-.016,-.01,-.004,.001,2*.002,.004,.012     DENSY      38
B,.018,.0215,.022,.022,.021,.017,.013,.0115,.011,.0105,6*.01/           DENSY      39
 DATA D1/ 2*.2086020E-3,-.1043010E-2,.4768047E-2,-.6898769E-1,          DENSY      40
A-.2881728E-1,-.2891209    ,-.1469915E-1,.3479176    ,-.1769713    ,    DENSY      41
B.3599678     ,-.6290010E-1, .8716393E-2,-.4442123E-2, .9052100E-2,     DENSY      42
C-.6592146E-1, .2546337    ,-.3526136    , .2806414E+1,-.1273043E+1,    DENSY      43
D .6159237    ,-.2349455E+1,-.3218101E+1,-.1578136E+1, .2330648E+1,     DENSY      44
E .1855542E+1, .4647183E+1,-.1244275E+1, .3294169    ,-.2475392E+1,     DENSY      45

F-.2834512E-1, .1887734    , .4073251E+1,-.2081778E+1,-.5461377    ,    DENSY      46
G-.1733670E+1, .2808207    ,-.5896124    ,-.3223710    , .6191931E-1,   DENSY      47
H .7469374E-1, .6279989    ,-.1866893    , .1187585    , .3706905E-1,   DENSY      48
I-.2735865E-2, .7321330E-3,-.1926665E-3, .3853331E-4,.3853331E-4/       DENSY      49
 DATA Y2/-5.,-4.,-3.5,-3.,-2.75,-2.5,-2.25,-2.,-1.75,-1.5,-1.25,        DENSY      50
A-1.1,-1.,-.95,-.9,-.85,-.8,-.6,-.5,-.4,-.3,-.25,-.2,-.15,-.1,-.05,     DENSY      51
B0.,.05,.1,.15,.2,.25,.3,.4,.5,.6,.7,.8,.9,1.,1.1,1.5,2.,2.5,5./        DENSY      52
 DATA Z2/4*-.025,-.022,-.019,-.013,-.008,3*-.004,-.0035,-.002,-.001     DENSY      53
A5,4*0.,.003,.006,-.002,-.01,-.02,-.022,-.013,-.001,.01,.017,.019,      DENSY      54
B.015,.007,.002,0.,-.002,-.004,-.006,3*-.008,-.006,-.003,.006,.016,     DENSY      55
C.026,.076/                                                             DENSY      56
 DATA D2/2*.1920707E-2,-.1536566E-1,.5954193E-1,-.3852028E-1,           DENSY      57
A .9453919E-1,-.5163651E-1, .1600686E-1,-.1083909    , .3355691E-1,     DENSY      58
B-.2583672E-1, .2152010    ,-.3372499    , .9930979    ,-.1235141E+1,   DENSY      59
C .3474687    ,-.1547332    , .2999660    , .3096702    ,-.1538646E+1,   DENSY      60
D-.7550822    ,-.1992212E+1, .3923932E+1, .5496481E+1, .4901429    ,    DENSY      61
E-.2570528    ,-.1861931E+1,-.1895221E+1,-.2557182E+1,-.2276048E+1,     DENSY      62
F .2061374E+1, .1230549E+1, .2164280    ,-.6455882E-1, .4180724E-1,     DENSY      63
G-.1026701    , .3688734    ,-.1728234    , .3224205    , .8314146E-1,   DENSY      64
H-.5498634E-1, .4180500E-2,-.1060723E-2, .6239550E-4, .6239550E-4/      DENSY      65
 DATA TITLE1/  4HFLAT,4H SID,4HEWAL,4HL DE,4HNSIT,4HY FI,4HELD ,        DENSY      66
14HBASE,4HD ON,4H SR=,4H1 DA,4HTA  ,8*4H    /                           DENSY      67
 DATA TITLE2/  4H   C,4HONTO,4HURED,4H SID,4HEWAL,4HL DE,4HNSIT,        DENSY      68
14HY FI,4HELD ,4HBASE,4HD ON,4H SR=,4H1 DA,4HTA  ,6*4H    /             DENSY      69
 DATA TITLE3/  4H LAT,4HEST ,4HTWO ,4H STA,4HGE D,4HENSI,4HTY F,        DENSY      70
14HIELD,4H LAM,4HINAR,4H - 2,4H PER,4HCENT,7*4H    /                    DENSY      71
 DATA Y3/ -3.546666, -2.435555, -1.324444, -0.435555, -0.324444,        DENSY      72
X         -0.257777, -0.235555, -0.224444, -0.220000, -0.215555,        DENSY      73
X         -0.213333, -0.206666, -0.202222, -0.191111, -0.180000,        DENSY      74
X         -0.175555, -0.171111, -0.164444, -0.157777, -0.146666,        DENSY      75
X         -0.124444, -0.113333, -0.102222, -0.091111, -0.080000,        DENSY      76
X         -0.068888, -0.057777, -0.035555, -0.013333, -0.002222,        DENSY      77
X          0.008888,  0.015555,  0.019999,  0.024444,  0.031111,        DENSY      78
X          0.042222,  0.053333,  0.064444,  0.075555,  0.120000,        DENSY      79
X          0.164444,  0.231111,  0.342222,  0.453333,  0.675555,        DENSY      80
X          0.897777,  1.342222,  2.008888,  2.453333,  3.120000/        DENSY      81
 DATA Z3/ 0.00, 0.00, 0.00, 0.00, 0.00, 0.00, 0.00, 0.02, 0.06,         DENSY      82
X         0.13, .185, 0.55, 0.87, 1.50, 1.97, 2.04, 2.06, 2.04,         DENSY      83
X         1.98, 1.85, 1.50, 1.27, 1.15, 1.10, 1.07, 0.98, 0.88,         DENSY      84
X         0.67, 0.47, 0.34, 0.16, 0.09, 0.05, 0.03, 0.01, 0.00,         DENSY      85
X         0.00, 0.00, 0.00, 0.00, 0.00, 0.00, 0.00, 0.00, 0.00,         DENSY      86
X         0.00, 0.00, 0.00, 0.00, 0.00/                                 DENSY      87
 DATA D3/    2*.291422E-02,-.145711E-01, .619272E-01,-.998122E+00,      DENSY      88
X .522010E+01,-.387664E+02, .119415E+04, .145780E+04, .208712E+04,      DENSY      89
X .886166E+04, .267319E+04,-.337096E+04, .107416E+03,-.483470E+04,      DENSY      90
X-.226811E+04,-.128032E+04,-.970171E+03,-.238990E+03,-.111126E+03,      DENSY      91
X-.640623E+03, .139299E+04, .414640E+03, .350441E+03,-.844406E+03,      DENSY      92
X .111185E+03,-.863363E+02, .819162E+02,-.119828E+03,-.902860E+03,      DENSY      93
X .130126E+04,-.305338E+03, .159978E+04,-.188176E+02, .346199E+03,      DENSY      94
X .374516E+02,-.100062E+02, .257350E+01,-.287744E+00, .759831E-01,      DENSY      95
X-.161884E-01, .330596E-02,-.866024E-03, .158135E-03,-.413956E-04,      DENSY      96
X .744651E-05,-.164175E-05, .508161E-06,-.781786E-07,-.781786E-07/      DENSY      97
 DATA TITLE8/4H    ,4HRUN ,4H112 ,4H1.6 ,4HSEC ,4HRIGH,                 SOQ77CY1   25
```

```
X   4HT ST,4HAGE ,4HEAST,4H AND,4HWEST,4H WAL,4HL AN,4HALYT,4HIC  ,  SOU77CY1    26
X   4H     ,4H     ,4H     ,4H     ,4H                                 SOU77CY1    27
 DATA TM8/.20345/,M8/22/,XM8/.0093/, M8W/22/                           SOU77CY1    28
 DATA Y8/-1.8,-1.7,-1.6,-1.5,-1.4,-1.3,-1.2,-1.,-.8,-.7,-.6,-.5,       SOU77CY1    29

X -.4,-.3,-.2,-.1,-.04,0,.06,.1,.2,.37/                                SOU77CY1    30
 DATA Z8/0.,.3,.8,1.1,1.2,1.8,2.4,2.6,3.1,3.6,3.9,3.6,3.3,3.8,5.4,     SOU77CY1    31
X 8.1,8.3,8.1,7.6,7.7,10.4,13.42/                                      SOU77CY1    32
 DATA Y8W/-2.,-1.9,-1.7,-1.6,-1.5,-1.4,-1.3,-1.2,-1.1,-1.,-.8,-.7,     SOU77CY1    33
X -.6,-.5,-.4,-.3,-.2,-.06,0.,.1,.2,.36/                               SOU77CY1    34
 DATA Z8W/0.,0.,.5,.8,1.1,1.1,.8,.1,-.4,-.9,-1.5,-1.7,-1.8,-1.7,       SOU77CY1    35
X -1.2,.3,2.3,2.4,3.5,6.,7.8,10./                                      SOU77CY1    36
 DATA TITLE9/4H     ,4HRUN ,4H109 ,4H1.8 ,4HSEC ,4HLEFT,               SOU77CY1    37
X   4H  ST,4HAGE ,4HEAST,4H AND,4HWEST,4H WAL,4HL AN,4HALYT,4HIC  ,    SOU77CY1    38
X   4H     ,4H     ,4H     ,4H     ,4H                                 SOU77CY1    39
 DATA TM9/.20345/,M9/16/,XM9/.01/, M9W/19/                             SOU77CY1    40
 DATA Y9/-1.76,-1.28,-1.2,-1.16,-.88,-.64,-.52,-.42,-.34,-.2,          SOU77CY1    41
X -.12,-.06,.08,.16,.42,.52/                                           SOU77CY1    42
 DATA Z9/-.3,1.1,1.1,1.,-.8,-1.6,-1.65,-1.3,-.6,1.4,1.7,1.35,0.,       SOU77CY1    43
X .27,3.4,5.3/                                                         SOU77CY1    44
 DATA Y9W/-1.76,-1.52,-1.4,-1.26,-1.2,-.96,-.66,-.44,-.37,-.3,         SOU77CY1    45
X -.2,-.1,-.06,.02,.05,.1,.32,.44,.62/                                 SOU77CY1    46
 DATA Z9W/-.3,.2,.9,1.7,1.6,.35,-.7,-1.2,-1.1,-.55,1.03,2.3,2.05,      SOU77CY1    47
X .45,.3,.5,2.5,3.7,6.8/                                               SOU77CY1    48
C********************************************************************* DENSY       98
      DATA BLANK/4H    /                                               DENSY       99
      M = YLEN                                                         DENSY      100
      XMACH=4.56                                                       DENSY      101
      LAG=FLAG+.1                                                      DENSY      102
      MUT = NPX*NPY                                                    DENSY      103
      DO 1629 IZERO=1,MUT                                              DENSY      104
 1629 P(IZERO) = 0.                                                    DENSY      105
C     CALL ZERO(P(1),P(MUT))                                           DENSY      106
      LL=1                                                             DENSY      107
      IF(LAG.EQ.5.OR.LAG.EQ.7.OR.LAG.EQ.8.OR.LAG.EQ.9) LL=-(NSYM-2)    SOU77CY1    49
      GO TO (100,200,300,400,500,2,500,800,901,1001,1001),LAG          SOU77CY1    50
C CONTOURED SIDEWALL DENSITY FIELD                                     DENSY      110
  100 TM(1)=TM2*SQRT((XM2**2-1.)/(XMACH**2-1.))                        DENSY      111
      XMULG=(XM2**2/SQRT(XM2**2-1.))/(XMACH**2/SQRT(XMACH**2-1.))      DENSY      112
      XMULT(1) = 1./XMULG                                              DENSY      113
      XLS(1)=0.0                                                       DENSY      114
      M(1) = 45                                                        DENSY      115
      DO 110  I=1,45                                                   DENSY      116
      Y(I,1)=Y2(I)                                                     DENSY      117
      Z(I,1)=Z2(I)                                                     DENSY      118
  110 D(I,1)=D2(I)                                                     DENSY      119
      DO 120 I=1,20                                                    DENSY      120
  120 TITLE(I)=TITLE2(I)                                               DENSY      121
      GO TO 2                                                          DENSY      122
C FLAT SIDEWALL DENSITY DIELD                                          DENSY      123
  200 TM(1)=TM1*SQRT((XM1**2-1.)/(XMACH**2-1.))                        DENSY      124
      XMULG=(XM1**2/SQRT(XM1**2-1.))/(XMACH**2/SQRT(XMACH**2-1.))      DENSY      125
      XMULT(1) = 1./XMULG                                              DENSY      126
      XLS(1)=0.0                                                       DENSY      127
      M(1) = 50                                                        DENSY      128
      DO 210 I=1,50                                                    DENSY      129
      Y(I,1)=Y1(I                                                      DENSY      130
      Z(I,1)=Z1(I)                                                     DENSY      131
  210 D(I,1)=D1(I)                                                     DENSY      132
      DO 220 I=1,20                                                    DENSY      133
  220 TITLE(I)=TITLE1(I)                                               DENSY      134
      GO TO 2                                                          DENSY      135

C LATEST AND GREATEST TWO STAGE DENSITY FIELD                          DENSY      136
  300 TM(1) = TM3                                                      DENSY      137
      XLS(1)=0.0                                                       DENSY      138
      XMULT(1) = XM3                                                   DENSY      139
      M(1) = M3                                                        DENSY      140
      DO 310 I = 1,M3                                                  DENSY      141
      Y(I,1) = Y3(I)                                                   DENSY      142
      Z(I,1) = Z3(I)                                                   DENSY      143
  310 D(I,1) = D3(I)                                                   DENSY      144
```

```
      DO 320 I = 1,20                                                   DENSY     145
  320 TITLE(I) =TITLE3(I)                                               DENSY     146
      GO TO 2                                                           DENSY     147
C******* ALL STAGE DENSITY FIELD (ANALYTICAL SIDEWALL PROJECTION) ****** SOQ77CY1   51
  800 IF (FLAG.LT.8.05) GO TO 802                                       SOQ77CY1   52
      READ (5,DENS8)                                                    SOQ77CY1   53
      READ (5,807) TITLE8                                               SOQ77CY1   54
  807 FORMAT(20A4)                                                      SOQ77CY1   55
  802 TM(1) =  TM8                                                      SOQ77CY1   56
      XSEED=7.                                                          SOQ77CY1   57
      XLS(1) = 0.0                                                      SOQ77CY1   58
      XMULT(1) = XM8                                                    SOQ77CY1   59
      M(1) = M8                                                         SOQ77CY1   60
      DO 810  I=1,M8                                                    SOQ77CY1   61
      Y(I,1) = Y8(I)                                                    SOQ77CY1   62
      Z(I,1) = Z8(I)                                                    SOQ77CY1   63
 810  D(I,1) = 0.0                                                      SOQ77CY1   64
      IF(LL.EQ.1) GO TO 815                                             SOQ77CY1   65
       TM(2) = TM8                                                      SOQ77CY1   66
      XLS(2) = 0.0                                                      SOQ77CY1   67
      XMULT(2) = XM8                                                    SOQ77CY1   68
      M(2) = M8W                                                        SOQ77CY1   69
      DO 811  I=1,M8W                                                   SOQ77CY1   70
      Y(I,2) = Y8W(I)                                                   SOQ77CY1   71
      Z(I,2) = Z8W(I)                                                   SOQ77CY1   72
  811 D(I,2) = 0.0                                                      SOQ77CY1   73
  815 DO 820 I=1,20                                                     SOQ77CY1   74
  820 TITLE(I) = TITLE8(I)                                              SOQ77CY1   75
      GO TO 2                                                           SOQ77CY1   76
  901 IF (FLAG.LT.9.05) GO TO 904                                       SOQ77CY1   77
      READ (5,DENS9)                                                    SOQ77CY1   78
      READ (5,807) TITLE9                                               SOQ77CY1   79
  904 TM(1) = TM9                                                       SOQ77CY1   80
      XSEED=7.                                                          SOQ77CY1   81
      XLS(1) = 0.0                                                      SOQ77CY1   82
      XMULT(1) = XM9                                                    SOQ77CY1   83
      M(1) = M9                                                         SOQ77CY1   84
      DO 910  I=1,M9                                                    SOQ77CY1   85
      Y(I,1) = Y9(I)                                                    SOQ77CY1   86
      Z(I,1) = Z9(I)                                                    SOQ77CY1   87
  910 D(I,1) = 0.0                                                      SOQ77CY1   88
      IF(LL.EQ.1) GO TO 915                                             SOQ77CY1   89
      TM(2) = TM9                                                       SOQ77CY1   90
      XLS(2) = 0.0                                                      SOQ77CY1   91
      XMULT(2) = XM9                                                    SOQ77CY1   92
      M(2) = M9W                                                        SOQ77CY1   93
      DO 911  I=1,M9W                                                   SOQ77CY1   94
      Y(I,2) = Y9W(I)                                                   SOQ77CY1   95
      Z(I,2) = Z9W(I)                                                   SOQ77CY1   96
 911  D(I,2) = 0.0                                                      SOQ77CY1   97
  915 DO 920 I=1,20                                                     SOQ77CY1   98
  920 TITLE(I) = TITLE9(I)                                              SOQ77CY1   99
      GO TO 2                                                           SOQ77CY1  100
  500 READ (IN,987) (TITLE(I),I=1,17)                                   DENSY     148
  987 FORMAT (17A4)                                                     DENSY     149
      DO 765 I = 1,3                                                    DENSY     150
  765 TITLE(17+I) = BLANK                                               DENSY     151
  989 FORMAT     (3F10.6,I5)                                            DENSY     152
  988 FORMAT (2F10.6,E13.6)                                             DENSY     153
    2 DO 503 L = 1,LL                                                   DENSY     154
      IF (LAG.NE.5.AND.LAG.NE.7)  GO TO 222                             DENSY     155
      READ (IN,989)    XLS(L),  XMULT(L),  TM(L),  M(L)                 DENSY     156
      MMM = M(L)                                                        DENSY     157
      DO 502 I = 1,MMM                                                  DENSY     158
  502 READ (IN,988) Y(I,L),Z(I,L),D(I,L)                                DENSY     159
C COMPUTE PHASE DISTORTION IN SEGMENT                                   DENSY     160
  222 WRITE(6,56) (TITLE(I),I=1,20)                                     DENSY     161
   56 FORMAT(1H1,2X,20A4)                                               DENSY     162
      WRITE(6,3) RHO,       H,    FLAG,XLS(L),XMULT(L),TM(L),M(L)       DENSY     163
    3 FORMAT(58H0  RHO            H             FLAG  XLS  XMULT   TM   DENSY     164
     1M /E10.3,5X,F7.3,7X,F5.1,2F6.3,F8.5,I3/17X,1H5,11X,6HDELRHO,6X,   DENSY     165
```

```
     211HCOEFFICIENT )
      MMM = M(L)
      WRITE(6,4) (Y(I,L),Z(I,L),D(I,L),I=1,MMM)
    4 FORMAT(10X,F10.5,5X,F10.5,4X,E14.7)
  503 RHOCL(L)=-RHO*GDC*ZSLAB*XMULT(L)
      DX=XLEN/NPX
      DY=YLEN/NPY/(NSYM+1)
      IZ=0
      DO 10 I=1,NPY
      S=DY*(I-.5)
      DO 10 J=1,NPX
      X=DX*(J-.5)
      IZ=IZ+1
      IF(LAG .EQ. 8.OR.LAG.EQ.9)CALL LINTERP(X,S,OP)
C     IF(X.GT.20.)WRITE(6,2051)X,S,OP,IZ
 2051  FORMAT(10X,9HX S OP IZ,3(5X,E15.7),I5)
      IF (LAG.LT.8) CALL ROSN(X,S,OP)
   10 P(IZ )=OP
C
      GO TO 1000
C (( MODIFIED 1/14/77 FAA TO READ 2 DENSITY FIELDS FROM DISK
C     FLAG=10. READS FIELD FROM UNIT 30
C     FLAG=11. READS FIELD FROM UNIT 31
 1001 IF(LAG.EQ.10)IDENS=30
      IF(LAG.EQ.11)IDENS=31
C ))
      NOB = MOT
      NOBB= NOB
      IF(NSYM.NE.0)WRITE(6,113)
  113 FORMAT(5X,43HERROR-DENSITY FIELD CHOSEN NOT COMMENSURATE ,
     X45HWITH SYMMETRIC MESH, PROGRAM STOP ENCOUNTERED ,/)
      IF(NSYM.NE.0)STOP
      IF(NOBB.NE.NOB)WRITE(6,112)
      IF(NOBB.NE.NOB)STOP

  112 FORMAT(5X,39HCURRENT MESH PTS NOT IN AGREEMENT WITH
     X,49HSTORED DENSITY VALUES,PROGRAM STOP IN DENSITY,PLZ
     Y,11HCHECK INPUT  / )
      PHASE =(ZSLAB/196.32)
      READ(IDENS)(P(IZ),IZ=1,NOBB)
C (( REVISED ON OR BEFORE 12/7/76 F.ADAMEK
      READ(IDENS)AVOPD
      REWIND IDENS
      WRITE(6,1986)AVOPD
 1986 FORMAT(34H AVOPD IN AREA OF CONVEX MIRROR =  ,E15.6)
      DO 955 KK = 1,NOBB
      P(KK) =(P(KK) -AVOPD)* PHASE
  955 CONTINUE
      WRITE(6,114)IDENS,NOBB
  114 FORMAT(5X,28HDENSITY FIELD READ FROM UNIT ,I3,2H, ,I5,
     A8HPTS READ  /)
 1000 CONTINUE
C  --PLOTPT=0. FOR NO PLOTTING POINTS
C  --PLOTPT=1. FOR PLOTTING POINTS IN X DIR. THRU CENTER OF CAVITY
C  --PLOTPT=2. FOR PLOTTING POINTS IN Y DIR. THRU CENTER OF BEAM
      PLOTPT=0.
      IF(PLOTPT.EQ.0.) GO TO 1236
      WRITE(6,1987)
 1987  FORMAT(//20X,15HPLOTTING POINTS
      III=0
      AUX=XLEN/NPX
      XUY=YLEN/NPY/(NSYM+1)
      DO 1235 I=1,NPY
      YYY=AUY*(I-.5)
      DO 1235 J=1,NPX
      XXX=AUX*(J-.5)
      III=III+1
      IF(PLOTPT.EQ.1.) GO TO 1989
      IF(XXX.LE.5.0.OR.XXX.GE.5.07)GO TO 1235
      GO TO 1990
 1989 IF(YYY.LE.5.6.OR.YYY.GE.5.7)GO TO 1235
 1990 WRITE(6,3456)XXX,YYY,P(III)
 3456 FORMAT(10X,17HX , Y , OPD/SEG   ,3(E15.7,5X))
```

```
 1235 CONTINUE                                                          SOU77CY1   158
C ))                                                                    SOU77CY1   159
 1236 CONTINUE                                                          SOU77CY1   160
C DRAW PICTURE OF PHASE SHIFT PER SEGMENT                               DENSY      184
      PMAX=P(1)                                                         DENSY      185
      PMIN=PMAX                                                         DENSY      186
      DO 50 I=1,MUT                                                     DENSY      187
      PMIN = AMIN1(PMIN,P( I ))                                         DENSY      188
   50 PMAX = AMAX1(PMAX,P( I ))                                         DENSY      189
      DP=PMAX-PMIN                                                      DENSY      190
      IF (LAG.LT.10) WRITE (6,51) TITLE                                 SOU77CY1   161
      IF (LAG.GE.10) WRITE (6,5263)                                     SOU77CY1   162
 5263 FORMAT(1H1)                                                       SOU77CY1   163
   51 FORMAT(1H1,25X,20A4)                                              DENSY      192
      INTE=1                                                            DENSY      193
      IF (NPX.GT.128) INTE=2                                            DENSY      194
      DO 52 J=1,NPY                                                     DENSY      195
      KJ= (NPY-J) * NPX                                                 DENSY      196
      DO 53 I=1,NPX,INTE                                                DENSY      197
   53 IP(I)=10.0*(1.0-(P(I+KJ)-PMIN)/DP)                                DENSY      198
   52 WRITE(6,54)(IP(I),I=1,NPX,INTE)                                   DENSY      199
   54 FORMAT(2X,130I1)                                                  DENSY      200
      WRITE(6,55) PMIN,PMAX,DP                                          DENSY      201
   55 FORMAT(13H0MIN VALUE IS,E15.7,5X,12HMAX VALUE IS,E15.7,5X,        DENSY      202
     137H NORMALIZING FACTOR FOR ABOVE PLOT IS ,E15.7)                  DENSY      203
      RETURN                                                            DENSY      204
C XLS-1 MOD 6 NOZZLES SPLINE DATA FROM FILE 14                          DENSY      205
  400 READ(IF,1400) TILE                                                DENSY      206
 1400 FORMAT(12A4)                                                      DENSY      207
      READ(IF,1401) N4,M4                                               DENSY      208
 1401 FORMAT(16I5)                                                      DENSY      209
      READ(IF,1402) X4,Y4,Z4,C4                                         DENSY      210
 1402 FORMAT(5E16.8)                                                    DENSY      211
      DO 401 I=1,12                                                     DENSY      212
  401 TITLE(I)=TILE(I)                                                  DENSY      213
      DO 402 I=13,20                                                    DENSY      214
  402 TITLE(I)=BLANK                                                    DENSY      215
      K=M4(1)                                                           DENSY      216
      H=Y4(1,K)-Y4(1,1)                                                 DENSY      217
      ROCL=-RHO*GDC*ZSLAB                                               DENSY      218
      DX=XLEN/NPX                                                       DENSY      219
      DY=YLEN/(NPY*(NSYM+1))                                            DENSY      220
      DO 410 I=1,NPX                                                    DENSY      221
      X=DX*(I-.5)                                                       DENSY      222
      DO 410 J=1,NPY                                                    DENSY      223
      S= DY*(J-.5) - (YLEN-8.) / 2.                                     DENSY      224
      CALL ROSN6(X,S,DP)                                                DENSY      225
  410 P(I+(J-1)*NPX) = DP                                               DENSY      226
      GO TO 1000                                                        SOU77CY1   164
      END                                                               DENSY      228
```

10. SUBROUTINE FOURT

a. Purpose -- Subroutine FOURT performs a forward or backward Fast Fourier Transform on any multidimensional complex array by efficiently performing the summation.

$$A_m = \sum_{N=0}^{N-1} X_n \, \varepsilon^{\pm 2\pi imn/N} \qquad (39)$$

The transform pair that needs to be evaluated is

$$F(s) = \int_{-\infty}^{\infty} f(x)\varepsilon^{2\pi ixs}\, dx \tag{40}$$

and

$$f(x) = \int_{-\infty}^{\infty} F(s)\varepsilon^{-2\pi ixs}\, ds \tag{41}$$

To digitally evaluate an integral, the continuous form of an integral must be changed to its discrete form. For example,

$$G = \int_a^b g(x)\, dx \Rightarrow \lim_{N\to\infty} \sum_{n=0}^{N} gn\Delta X \tag{42}$$

b. Relevant formalism -- Assume that all the intervals, ΔX_n, are chosen to be equal and that the infinite sum can be approximated by a finite sum. Then,

$$G \approx \sum_{n=0}^{N-1} g_n \, (x_{n+1} - x_n) \text{ with } g_n = g\left(x = \frac{n(b-a)}{N}\right) \tag{43}$$

$$G = \sum_{n=0}^{N-1} g_n \left[(n+1)\ \frac{(b-a)}{N}\ -n\ \frac{(b-a)}{N}\right]$$

or

$$G = \Delta X \sum_{n=0}^{N-1} g_n \int_a^b g(x)\, dx \tag{44}$$

To evaluate Equations (40) and (41) by the approximate form (Eq. (44)), assume that the function f(x) is spatially bounded in $0 \le x < 2L$ and that it is a band-limited function so that F(s) is confined in the region $-B \le S \le B$. To perform either a backward or forward Fourier transform, the functions f and F should differ in form only by the sign of the exponent. Therefore, the properties of F must be evaluated so that its region can be changed to $0 \le S \le 2B$. This is easily done by replicating the function f(x) so that it is periodic with period 2L. This will not change the value of f in the region of interest and, by proper choice of N, will return the desired function F.

A sampled function, f_s, can be analytically represented by a Dirac delta function:

$$f_s(x) = \sum_{n=0}^{N-1} f_n \; \delta(x - n\Delta x) \text{ with } \Delta x = \frac{2L}{N} \tag{45}$$

A replicated function can be represented by a convolution:

$$\begin{aligned} f_{rep}(x) &= \int_0^{2L} dx' \, f(x) \sum_{n=-\infty}^{\infty} \delta\left(x - (x' + 2LN)\right) \\ &= f(x) \circledast \sum_{n=-\infty}^{\infty} \delta(x - n2L) \end{aligned} \tag{46}$$

Therefore, a sampled and replicated function is represented by:

$$\hat{f}(x) = \sum_{n=0}^{N-1} f_n \; \delta(x - n\Delta x) \circledast \sum_{m=-\infty}^{\infty} \delta(x - nN\Delta x) \tag{47}$$

The Fourier Transform $\hat{F}(s)$ of $\hat{f}(x)$ is

$$\hat{F}(s) = \mathcal{F}\{\hat{f}\} = \mathcal{F}\left\{\sum_{n=0}^{N-1} f_n \; \delta(x - n\Delta x)\right\} \mathcal{F}\left\{\sum_{m=-\infty}^{\infty} \delta(x - nN\Delta x)\right\} \tag{48}$$

by the convolution theorem. Since

$$\sum_{n=-\infty}^{\infty} \delta(x-na) = \frac{1}{a} \sum_{n=-\infty}^{\infty} \varepsilon^{2\pi i n \frac{x}{a}} \tag{49}$$

one finds,

$$\hat{F}(s) = \sum_{n=0}^{N-1} f_n \varepsilon^{2\pi i s n \Delta x} \sum_{m=-\infty}^{\infty} \frac{1}{N\Delta x} \delta\left(s - \frac{n}{N\Delta x}\right) \tag{50}$$

Rearranged this gives

$$\hat{F}(s) = \frac{1}{N\Delta x} \sum_{m=-\infty}^{\infty} \delta\left(s - \frac{m}{N\Delta x}\right) \sum_{n=0}^{N-1} f_n \varepsilon^{2\pi i m n/N} \tag{51}$$

Recalling Equations (40) and (44), define

$$F_n = \Delta x \sum_{n=0}^{N-1} f_n \varepsilon^{2\pi i n m/N} = F_{n+N} \tag{52}$$

Then

$$\hat{F}(s) = \frac{1}{N(\Delta x)^2} \sum_{m=-\infty}^{\infty} F_m \ \delta\left(s - \frac{m}{N\Delta x}\right) \tag{53}$$

Since $F_m = F_{m+n}$, one can rewrite the above as a replication for every N point.

$$\hat{F}(s) = \frac{1}{N(\Delta x)^2} \sum_{m=0}^{N-1} F_m \ \delta\left(s - \frac{m}{N\Delta x}\right) \sum_{n=-\infty}^{\infty} \delta\left(s - \frac{n}{\Delta x}\right) \tag{54}$$

Therefore, by replicating f(x) with period 2L, F is periodic with period $1/\Delta X$.

So by choosing N so that $N/2L \geq 2B$, rewrite the limits for F as $0 \leq S \leq B$.

Since

$$\delta_{nk} = \frac{1}{N} \sum_{m=0}^{N-1} \varepsilon^{2\pi im(n-k)/n} = \begin{cases} 1, & n = k \\ 0, & n \neq k \end{cases} \tag{55}$$

invert (52) to find

$$f_n = \frac{1}{N\Delta x} \sum_{m=0}^{N-1} F_m \, \varepsilon^{-2\pi imn/N} \tag{56}$$

Thus, choosing $\Delta s = 1/N\Delta x$, the transform pair becomes

$$F_m = \Delta x \sum_{n=0}^{N-1} f_n \, \varepsilon^{-2\pi imn/N} \tag{57}$$

$$f_n = \Delta s \sum_{m=0}^{N-1} F_m \, \varepsilon^{-2\pi imn/N} \qquad (\Delta x \Delta s = \frac{1}{N}) \tag{58}$$

where, with $N/2L \geq 2B$, F_m represents $F(s)$ for $0 \leq S_m \leq 2B$ ($S_m = m\Delta s$) and f_n represents $f(x)$ for $0 \leq x_n \leq 2L$ ($x_n = n\Delta x$).

The transform pair f_n and F_m are now in a form usable by the Fast Fourier Transform (FFT). The FFT evaluates the sum

$$A_r = \sum_{k=0}^{N-1} X_k \, \varepsilon^{\pm 2\pi irk/N} \tag{59}$$

Following Higgins (Ref. 9), this sum can be split into two sums (choosing the + sign in the exponent):

$$A_r = \sum_{\substack{k=0 \\ (k\,\mathrm{even})}}^{N-1} X_k \, \epsilon^{\pi irk/N} + \sum_{\substack{k=0 \\ (k\,\mathrm{odd})}}^{N-1} X_k r^{2\pi irk/N} \tag{60}$$

Let

$$k = 0,\ 1,\ 3,\ 5,\ \ldots\ \frac{N}{2} - 1 \tag{61}$$

then

$$A_r = \sum_{k=0}^{\frac{N}{2}-1} \left[y_k \epsilon^{2\pi ir2k/N} + z_k \epsilon^{2\pi ir(2k+1)/N} \right] \tag{62}$$

Letting

$$B_r \equiv \sum_{k=0}^{\frac{N}{2}-1} y_k \epsilon^{4\pi irk/N} \tag{63}$$

and

$$C_r \equiv \sum_{k=0}^{\frac{N}{2}-1} z_k \epsilon^{4\pi irk/N} \tag{64}$$

---

9. Wiggins, R.J., "Fast Fourier Transform: An Introduction With Some Minicomputer Experiments," AJP, 44, 1976.

A_r can be written

$$A_r = B_r + C_r \epsilon^{2\pi i r/N} \tag{65}$$

Define

$$W_n \equiv \epsilon^{2\pi i/N} \tag{66}$$

Then,

$$A_r = B_r + C_r + (W_n)^r \tag{67}$$

By letting $r \to r + N/2$:

$$A\left(r + \frac{N}{2}\right) = B_r - (W_n)^r C_r \tag{68}$$

Therefore, A_r can be evaluated by doing two sums, each containing N/2 terms. However, these sums need to be performed for only half the r's $\left(0 \leq r < \frac{N}{2}\right)$ since $A_r + N/2$ is found using the two sums used in the evaluation of A_r. By initially forcing N to be a power of two by completing the array to be transformed with zeros, continue to divide each successful sum into two, until a "sum" is reduced to just one number, taking care to note that N changes with each division. When using the FFT, care must be taken to scale the output correctly since the FFT evaluates only sums of the form

$$A_r = \sum_{n=0}^{N-1} x_n \, \epsilon^{\pm 2\pi i n r/N} \tag{69}$$

and as can be seen from Equations (58) the Fourier Transforms contain Δx or Δs: If only forward then backward transforming is done, it is sufficient to divide the final answer by N for each dimension as is indicated by the last part of Equation (58).

Note that when the data are returned from the FFT the first data point is either the $x = 0$ or the $s = 0$ point. To see the actual frequency space pictures, assume a two-dimensional case. An isointensity printer plot of FFT output in frequency space might look like that shown in Figure 22.

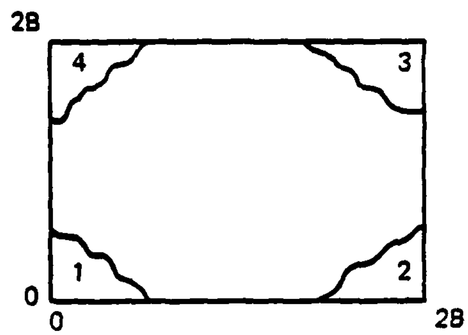


Figure 22. Example of isointensity printer plot of FFT output in frequency space.

To see the -B to +B version, the adjacent cells shown in Figure 23 must be added to Figure 22.

The subroutine FOURT computer printouts follow.

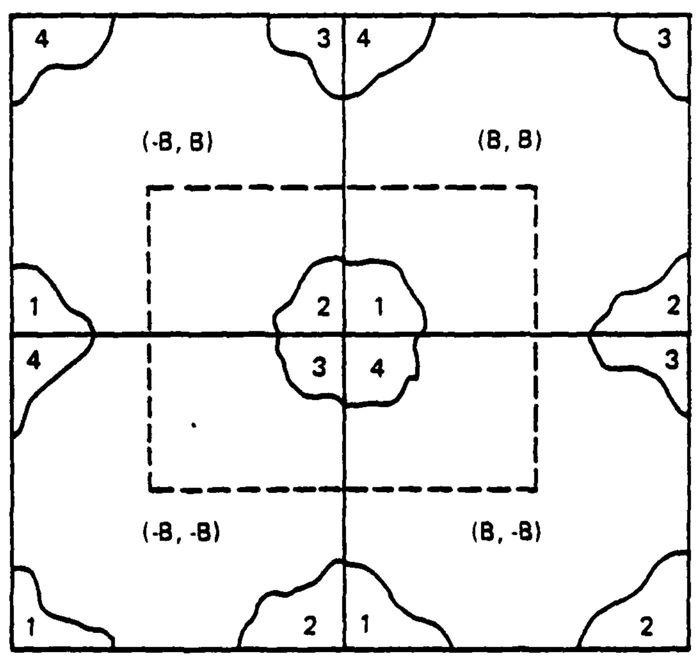


Figure 23. -B to +B version of isointensity printer plot of FFT output in frequency space.

```
      SUBROUTINE FOURT(DATA,NAR,NN,ISIGN)
C*****************************************************************C
C     THE COOLEY-TUKEY FAST FOURIER TRANSFORM IN USASI BASIC FORTRAN
C     TRANSFORM(K1,K2,...) = SUM(DATA(J1,J2,...)*EXP(ISIGN*2*PI*SQRT(-1)
C     *((J1-1)*(K1-1)/NN(1)+(J2-1)*(K2-1)/NN(2)+...))), SUMMED FOR ALL
C     J1, K1 BETWEEN 1 AND NN(1), J2, K2 BETWEEN 1 AND NN(2), ETC.
C     THERE IS NO LIMIT TO THE NUMBER OF SUBSCRIPTS.  DATA IS A
C     MULTIDIMENSIONAL COMPLEX ARRAY WHOSE REAL AND IMAGINARY
C     PARTS ARE ADJACENT IN STORAGE, SUCH AS FORTRAN IV PLACES THEM.
C     IF ALL IMAGINARY PARTS ARE ZERO (DATA ARE DISGUISED REAL), SET
C     IFORM TO ZERO TO CUT THE RUNNING TIME BY UP TO FORTY PERCENT.
C     OTHERWISE, IFORM = +1.  THE LENGTHS OF ALL DIMENSIONS ARE
C     STORED IN ARRAY NN, OF LENGTH NDIM.  THEY MAY BE ANY POSITIVE
C     INTEGERS, THO THE PROGRAM RUNS FASTER ON COMPOSITE INTEGERS, AND
C     ESPECIALLY FAST ON NUMBERS RICH IN FACTORS OF TWO.  ISIGN IS +1
C     OR -1.  IF A -1 TRANSFORM IS FOLLOWED BY A +1 ONE (OR A +1
C     BY A -1) THE ORIGINAL DATA REAPPEAR, MULTIPLIED BY NTOT (=NN(1)*
C     NN(2)*...).  TRANSFORM VALUES ARE ALWAYS COMPLEX, AND ARE RETURNED
C     IN ARRAY DATA, REPLACING THE INPUT.  IN ADDITION, IF ALL
C     DIMENSIONS ARE NOT POWERS OF TWO, ARRAY WORK MUST BE SUPPLIED,
C     COMPLEX OF LENGTH EQUAL TO THE LARGEST NON 2**K DIMENSION.
C     OTHERWISE, REPLACE WORK BY ZERO IN THE CALLING SEQUENCE.
C     NORMAL FORTRAN DATA ORDERING IS EXPECTED, FIRST SUBSCRIPT VARYING
C     FASTEST.  ALL SUBSCRIPTS BEGIN AT ONE.
      LEVEL 2, DATA
      DIMENSION DATA(NAR),NN(2),IFACT(32),WORK(300)
      NDIM=2
      IFORM=+1
      WI=1.00
      WR=1.00
      WSTPR=1.00
      WSTPI=1.00
      TWOPI=6.283185307
      IF(NDIM-1)920,1,1
1     NTOT=2
      DO 2 IDIM=1,NDIM
      IF(NN(IDIM))920,920,2
2     NTOT=NTOT*NN(IDIM)
      NP1=2
      DO 910 IDIM=1,NDIM
      N=NN(IDIM)
      NP2=NP1*N
      IF(N-1)920,900,5
5     M=N
      NTWO=NP1
      IF=1
      IDIV=2
10    IQUOT=M/IDIV
      IREM=M-IDIV*IQUOT
      IF(IQUOT-IDIV)50,11,11
11    IF(IREM)20,12,20
12    NTWO=NTWO+NTWO
      M=IQUOT
      GO TO 10
20    IDIV=3
30    IQUOT=M/IDIV
      IREM=M-IDIV*IQUOT
      IF(IQUOT-IDIV)60,31,31
31    IF(IREM)40,32,40
32    IFACT(IF)=IDIV
      IF=IF+1
      M=IQUOT
      GO TO 30
40    IDIV=IDIV+2
      GO TO 30
50    IF(IREM)60,51,60
51    NTWO=NTWO+NTWO
      GO TO 70
```

```
60    IFACT(IF)=M
70    NON2=NP1*(NP2/NTWO)
      ICASE=1
      IF(IDIM-4)71,90,90
71    IF(IFORM)72,72,90
72    ICASE=2
      IF(IDIM-1)73,73,90
73    ICASE=3
      IF(NTWO-NP1)90,90,74
74    ICASE=4
      NTWO=NTWO/2
      N=N/2
      NP2=NP2/2
      NTOT=NTOT/2
      I=3
      DO 80 J=2,NTOT
      DATA(J)=DATA(I)
80    I=I+2
90    I1RNG=NP1
      IF(ICASE-2)100,95,100
95    I1RNG=NP0*(1+NPREV/2)
100   IF(NTWO-NP1)600,600,110
110   NP2HF=NP2/2
      J=1
      DO 150 I2=1,NP2,NON2
      IF(J-I2)120,130,130
120   I1MAX=I2+NON2-2
      DO 125 I1=I2,I1MAX,2
      DO 125 I3=I1,NTOT,NP2
      J3=J+I3-I2
      TEMPR=DATA(I3)
      TEMPI=DATA(I3+1)
      DATA(I3)=DATA(J3)
      DATA(I3+1)=DATA(J3+1)
      DATA(J3)=TEMPR
125   DATA(J3+1)=TEMPI
130   M=NP2HF
140   IF(J-M)150,150,145
145   J=J-M
      M=M/2
      IF(M-NON2)150,140,140
150   J=J+M
      NON2T=NON2+NON2
      IPAR=NTWO/NP1
310   IF(IPAR-2)350,330,320
320   IPAR=IPAR/4
      GO TO 310
330   DO 340 I1=1,I1RNG,2
      DO 340 J3=I1,NON2,NP1
      DO 340 K1=J3,NTOT,NON2T
      K2=K1+NON2
      TEMPR=DATA(K2)
      TEMPI=DATA(K2+1)
      DATA(K2)=DATA(K1)-TEMPR
      DATA(K2+1)=DATA(K1+1)-TEMPI
      DATA(K1)=DATA(K1)+TEMPR
340   DATA(K1+1)=DATA(K1+1)+TEMPI
350   MMAX=NON2
360   IF(MMAX-NP2HF)370,600,600
370   LMAX=MAX0(NON2T,MMAX/2)
      IF(MMAX-NON2)405,405,380
380   THETA=-TWOPI*FLOAT(NON2)/FLOAT(4*MMAX)
      IF(ISIGN)400,390,390
390   THETA=-THETA
400   WR=COS(THETA)
      WI=SIN(THETA)
      WSTPR=-2.*WI*WI
      WSTPI=2.*WR*WI
405   DO 570 L=NON2,LMAX,NON2T
      M=L
      IF(MMAX-NON2)420,420,410
```

```
410   W2R=WR*WR-WI*WI
      W2I=2.*WR*WI
      W3R=W2R*WR-W2I*WI
      W3I=W2R*WI+W2I*WR
420   DO 530 I1=1,I1RNG,2
      DO 530 J3=I1,NON2,NP1
      KMIN=J3+IPAR*M
      IF(MMAX-NON2)430,430,440
430   KMIN=J3
440   KDIF=IPAR*MMAX
450   KSTEP=4*KDIF
      DO 520 K1=KMIN,NTOT,KSTEP
      K2=K1+KDIF
      K3=K2+KDIF
      K4=K3+KDIF
      IF(MMAX-NON2)460,460,480
460   U1R=DATA(K1)+DATA(K2)
      U1I=DATA(K1+1)+DATA(K2+1)
      U2R=DATA(K3)+DATA(K4)
      U2I=DATA(K3+1)+DATA(K4+1)
      U3R=DATA(K1)-DATA(K2)
      U3I=DATA(K1+1)-DATA(K2+1)
      IF(ISIGN)470,475,475
470   U4R=DATA(K3+1)-DATA(K4+1)
      U4I=DATA(K4)-DATA(K3)
      GO TO 510
475   U4R=DATA(K4+1)-DATA(K3+1)
      U4I=DATA(K3)-DATA(K4)
      GO TO 510
480   T2R=W2R*DATA(K2)-W2I*DATA(K2+1)
      T2I=W2R*DATA(K2+1)+W2I*DATA(K2)
      T3R=WR*DATA(K3)-WI*DATA(K3+1)
      T3I=WR*DATA(K3+1)+WI*DATA(K3)
      T4R=W3R*DATA(K4)-W3I*DATA(K4+1)
      T4I=W3R*DATA(K4+1)+W3I*DATA(K4)
      U1R=DATA(K1)+T2R
      U1I=DATA(K1+1)+T2I
      U2R=T3R+T4R
      U2I=T3I+T4I
      U3R=DATA(K1)-T2R
      U3I=DATA(K1+1)-T2I
      IF(ISIGN)490,500,500
490   U4R=T3I-T4I
      U4I=T4R-T3R
      GO TO 510
500   U4R=T4I-T3I
      U4I=T3R-T4R
510   DATA(K1)=U1R+U2R
      DATA(K1+1)=U1I+U2I
      DATA(K2)=U3R+U4R
      DATA(K2+1)=U3I+U4I
      DATA(K3)=U1R-U2R
      DATA(K3+1)=U1I-U2I
      DATA(K4)=U3R-U4R
520   DATA(K4+1)=U3I-U4I
      KMIN=4*(KMIN-J3)+J3
      KDIF=KSTEP
      IF(KDIF-NP2)450,530,530
530   CONTINUE
      M=MMAX-M
      IF(ISIGN)540,550,550
540   TEMPR=WR
      WR=-WI
      WI=-TEMPR
      GO TO 560
550   TEMPR=WR
      WR=WI
      WI=TEMPR
560   IF(M-LMAX)565,565,410
565   TEMPR=WR
      WR=WR*WSTPR-WI*WSTPI+WR
```

```
570   WI=WI*WSTPR+TEMPR*WSTPI+WI
      IPAR=3-IPAR
      MMAX=MMAX+MMAX
      GO TO 360
600   IF(NTWO-NP2)605,700,700
605   IFP1=NON2
      IF=1
      NP1HF=NP1/2
610   IFP2=IFP1/IFACT(IF)
      J1RNG=NP2
      IF(ICASE-3)612,611,612
611   J1RNG=(NP2+IFP1)/2
      J2STP=NP2/IFACT(IF)
      J1RG2=(J2STP+IFP2)/2
612   J2MIN=1+IFP2
      IF(IFP1-NP2)615,640,640
615   DO 635 J2=J2MIN,IFP1,IFP2
      THETA=-TWOPI*FLOAT(J2-1)/FLOAT(NP2)
      IF(ISIGN)625,620,620
620   THETA=-THETA
625   SINTH=SIN(THETA/2.)
      WSTPR=-2.*SINTH*SINTH
      WSTPI=SIN(THETA)
      WR=WSTPR+1.
      WI=WSTPI
      J1MIN=J2+IFP1
      DO 635 J1=J1MIN,J1RNG,IFP1
      I1MAX=J1+I1RNG-2
      DO 630 I1=J1,I1MAX,2
      DO 630 I3=I1,NTOT,NP2
      J3MAX=I3+IFP2-NP1
      DO 630 J3=I3,J3MAX,NP1
      TEMPR=DATA(J3)
      DATA(J3)=DATA(J3)*WR-DATA(J3+1)*WI
630   DATA(J3+1)=TEMPR*WI+DATA(J3+1)*WR
      TEMPR=WR
      WR=WR*WSTPR-WI*WSTPI+WR
635   WI=TEMPR*WSTPI+WI*WSTPR+WI
640   THETA=-TWOPI/FLOAT(IFACT(IF))
      IF(ISIGN)650,645,645
645   THETA=-THETA
650   SINTH=SIN(THETA/2.)
      WSTPR=-2.*SINTH*SINTH
      WSTPI=SIN(THETA)
      KSTEP=2*N/IFACT(IF)
      KRANG=KSTEP*(IFACT(IF)/2)+1
      DO 698 I1=1,I1RNG,2
      DO 698 I3=I1,NTOT,NP2
      DO 690 KMIN=1,KRANG,KSTEP
      J1MAX=I3+J1RNG-IFP1
      DO 680 J1=I3,J1MAX,IFP1
      J3MAX=J1+IFP2-NP1
      DO 680 J3=J1,J3MAX,NP1
      J2MAX=J3+IFP1-IFP2
      K=KMIN+(J3-J1+(J1-I3)/IFACT(IF))/NP1HF
      IF(KMIN-1)655,655,665
655   SUMR=0.
      SUMI=0.
      DO 660 J2=J3,J2MAX,IFP2
      SUMR=SUMR+DATA(J2)
660   SUMI=SUMI+DATA(J2+1)
      WORK(K)=SUMR
      WORK(K+1)=SUMI
      GO TO 680
665   KCONJ=K+2*(N-KMIN+1)
      J2=J2MAX
      SUMR=DATA(J2)
      SUMI=DATA(J2+1)
      OLDSR=0.
      OLDSI=0.
      J2=J2-IFP2
```

```
670   TEMPR=SUMR                                              FOURT
      TEMPI=SUMI                                              FOURT
      SUMR=TWOWR*SUMR-OLDSR+DATA(J2)                          FOURT
      SUMI=TWOWR*SUMI-OLDSI+DATA(J2+1)                        FOURT
      OLDSR=TEMPR                                             FOURT
      OLDSI=TEMPI                                             FOURT
      J2=J2-IFP2                                              FOURT
      IF(J2-J3)675,675,670                                    FOURT
675   TEMPR=WR*SUMR-OLDSR+DATA(J2)                            FOURT
      TEMPI=WI*SUMI                                           FOURT
      WORK(K)=TEMPR-TEMPI                                     FOURT
      WORK(KCONJ)=TEMPR+TEMPI                                 FOURT
      TEMPR=WR*SUMI-OLDSI+DATA(J2+1)                          FOURT
      TEMPI=WI*SUMR                                           FOURT
      WORK(K+1)=TEMPR+TEMPI                                   FOURT
      WORK(KCONJ+1)=TEMPR-TEMPI                               FOURT
680   CONTINUE                                                FOURT
      IF(KMIN-1)685,685,686                                   FOURT
685   WR=WSTPR+1.                                             FOURT
      WI=WSTPI                                                FOURT
      GO TO 690                                               FOURT
686   TEMPR=WR                                                FOURT
      WR=WR*WSTPR-WI*WSTPI+WR                                 FOURT
      WI=TEMPR*WSTPI+WI*WSTPR+WI                              FOURT
690   TWOWR=WR+WR                                             FOURT
      IF(ICASE-3)692,691,692                                  FOURT
691   IF(IFP1-NP2)695,692,692                                 FOURT
692   K=1                                                     FOURT
      I2MAX=I3+NP2-NP1                                        FOURT
      DO 693 I2=I3,I2MAX,NP1                                  FOURT
      DATA(I2)=WORK(K)                                        FOURT
      DATA(I2+1)=WORK(K+1)                                    FOURT
693   K=K+2                                                   FOURT
      GO TO 698                                               FOURT
695   J3MAX=I3+IFP2-NP1                                       FOURT
      DO 697 J3=I3,J3MAX,NP1                                  FOURT
      J2MAX=J3+NP2-J2STP                                      FOURT
      DO 697 J2=J3,J2MAX,J2STP                                FOURT
      J1MAX=J2+J1RG2-IFP2                                     FOURT
      J1CNJ=J3+J2MAX+J2STP-J2                                 FOURT
      DO 697 J1=J2,J1MAX,IFP2                                 FOURT
      K=1+J1-I3                                               FOURT
      DATA(J1)=WORK(K)                                        FOURT
      DATA(J1+1)=WORK(K+1)                                    FOURT
      IF(J1-J2)697,697,696                                    FOURT
696   DATA(J1CNJ)=WORK(K)                                     FOURT
      DATA(J1CNJ+1)=-WORK(K+1)                                FOURT
697   J1CNJ=J1CNJ-IFP2                                        FOURT
698   CONTINUE                                                FOURT
      IF=IF+1                                                 FOURT
      IFP1=IFP2                                               FOURT
      IF(IFP1-NP1)700,700,610                                 FOURT
700   GO TO (900,800,900,701),ICASE                           FOURT
701   NHALF=N                                                 FOURT
      N=N+N                                                   FOURT
      THETA=-TWOPI/FLOAT(N)                                   FOURT
      IF(ISIGN)703,702,702                                    FOURT
702   THETA=-THETA                                            FOURT
703   SINTH=SIN(THETA/2.)                                     FOURT
      WSTPR=-2.*SINTH*SINTH                                   FOURT
      WSTPI=SIN(THETA)                                        FOURT
      WR=WSTPR+1.                                             FOURT
      WI=WSTPI                                                FOURT
      IMIN=3                                                  FOURT
      JMIN=2*NHALF-1                                          FOURT
      GO TO 725                                               FOURT
710   J=JMIN                                                  FOURT
      DO 720 I=IMIN,NTOT,NP2                                  FOURT
      SUMR=(DATA(I)+DATA(J))/2.                               FOURT
      SUMI=(DATA(I+1)+DATA(J+1))/2.                           FOURT
      DIFR=(DATA(I)-DATA(J))/2.                               FOURT
      DIFI=(DATA(I+1)-DATA(J+1))/2.                           FOURT
```

```
      TEMPR=WR*SUMI+WI*DIFR
      TEMPI=WI*SUMI-WR*DIFR
      DATA(I)=SUMR+TEMPR
      DATA(I+1)=DIFI+TEMPI
      DATA(J)=SUMR-TEMPR
      DATA(J+1)=-DIFI+TEMPI
  720 J=J+NP2
      IMIN=IMIN+2
      JMIN=JMIN-2
      TEMPR=WR
      WR=WR*WSTPR-WI*WSTPI+WR
      WI=TEMPR*WSTPI+WI*WSTPR+WI
  725 IF(IMIN-JMIN)710,730,740
  730 IF(ISIGN)731,740,740
  731 DO 735 I=IMIN,NTOT,NP2
  735 DATA(I+1)=-DATA(I+1)
  740 NP2=NP2+NP2
      NTOT=NTOT+NTOT
      J=NTOT+1
      IMAX=NTOT/2+1
  745 IMIN=IMAX-2*NHALF
      I=IMIN
      GO TO 755
  750 DATA(J)=DATA(I)
      DATA(J+1)=-DATA(I+1)
  755 I=I+2
      J=J-2
      IF(I-IMAX)750,760,760
  760 DATA(J)=DATA(IMIN)-DATA(IMIN+1)
      DATA(J+1)=0.
      IF(I-J)770,780,780
  765 DATA(J)=DATA(I)
      DATA(J+1)=DATA(I+1)
  770 I=I-2
      J=J-2
      IF(I-IMIN)775,775,765
  775 DATA(J)=DATA(IMIN)+DATA(IMIN+1)
      DATA(J+1)=0.
      IMAX=IMIN
      GO TO 745
  780 DATA(1)=DATA(1)+DATA(2)
      DATA(2)=0.
      GO TO 900
  800 IF(I1RNG-NP1)805,900,900
  805 DO 860 I3=1,NTOT,NP2
      I2MAX=I3+NP2-NP1
      DO 860 I2=I3,I2MAX,NP1
      IMIN=I2+I1RNG
      IMAX=I2+NP1-2
      JMAX=2*I3+NP1-IMIN
      IF(I2-I3)820,820,810
  810 JMAX=JMAX+NP2
  820 IF(IDIM-2)850,850,830
  830 J=JMAX+NP0
      DO 840 I=IMIN,IMAX,2
      DATA(I)=DATA(J)
      DATA(I+1)=-DATA(J+1)
  840 J=J-2
  850 J=JMAX
      DO 860 I=IMIN,IMAX,NP0
      DATA(I)=DATA(J)
      DATA(I+1)=-DATA(J+1)
  860 J=J-NP0
  900 NP0=NP1
      NP1=NP2
  910 NPREV=N
  920 CONTINUE
      RETURN
      END
```

11. SUBROUTINE FUHS

a. Purpose -- Subroutine FUHS is used to calculate the phase change due to heat release as the molecules in the lower laser level decay to the ground state, assuming supersonic flow and that the heat release has a disturbing effect (not major) on the flow. Figure 24 shows the subroutine FUHS flow chart.

b. Relevant formalism -- The equations used are based on those by Biblarz and Fuhs, (Ref. 10), and by Fuhs (Ref. 11).

Initially, it is assumed that the continuity, momentum, and energy equations for steady flow with heat addition are valid:

Continuity: $$\nabla \cdot (\rho\vec{u}) = 0 \tag{70}$$

Momentum: $$\rho\frac{D\vec{u}}{Dt} + \vec{\nabla}_p = 0 \tag{71}$$

Energy: $$\nabla \cdot \rho\vec{u}\left(h + \frac{\vec{u}^2}{2}\right) = q \tag{72}$$

These are linearized, assuming

$$\rho = \rho_\infty + \rho' \qquad p = p_\infty + p' \qquad \vec{u} = \hat{i}\,(U+u') + \hat{j}\,v' \tag{73}$$

resulting in

Continuity: $$\rho_\infty u'_x + \rho_\infty U'_y + U\rho'_x = 0 \tag{74}$$

$$\left(u' \equiv \frac{\partial}{\partial_x}u' \;;\ \text{etc.}\right) \tag{75}$$

Momentum: $$\begin{Bmatrix} \rho_\infty\, Uu'_x + p'_x = 0 \\ \rho_\infty\, Uv'_x + p'_y = 0 \end{Bmatrix} \tag{76}$$

10. Biblarz, O. and Fuhs, A. E., "Laser Cavity Density Changes with Kinetics of Energy Release," AIAA Journal, 12, p. 1083, August 1974.

11. Fuhs, A. E., "Quaside Area Rule for Heat Addition in Transonic and Supersonic Flight Regimes," AFAPL-TR-72-10, Air Force Propulsion Laboratory, WPAFB, Ohio, 1972.

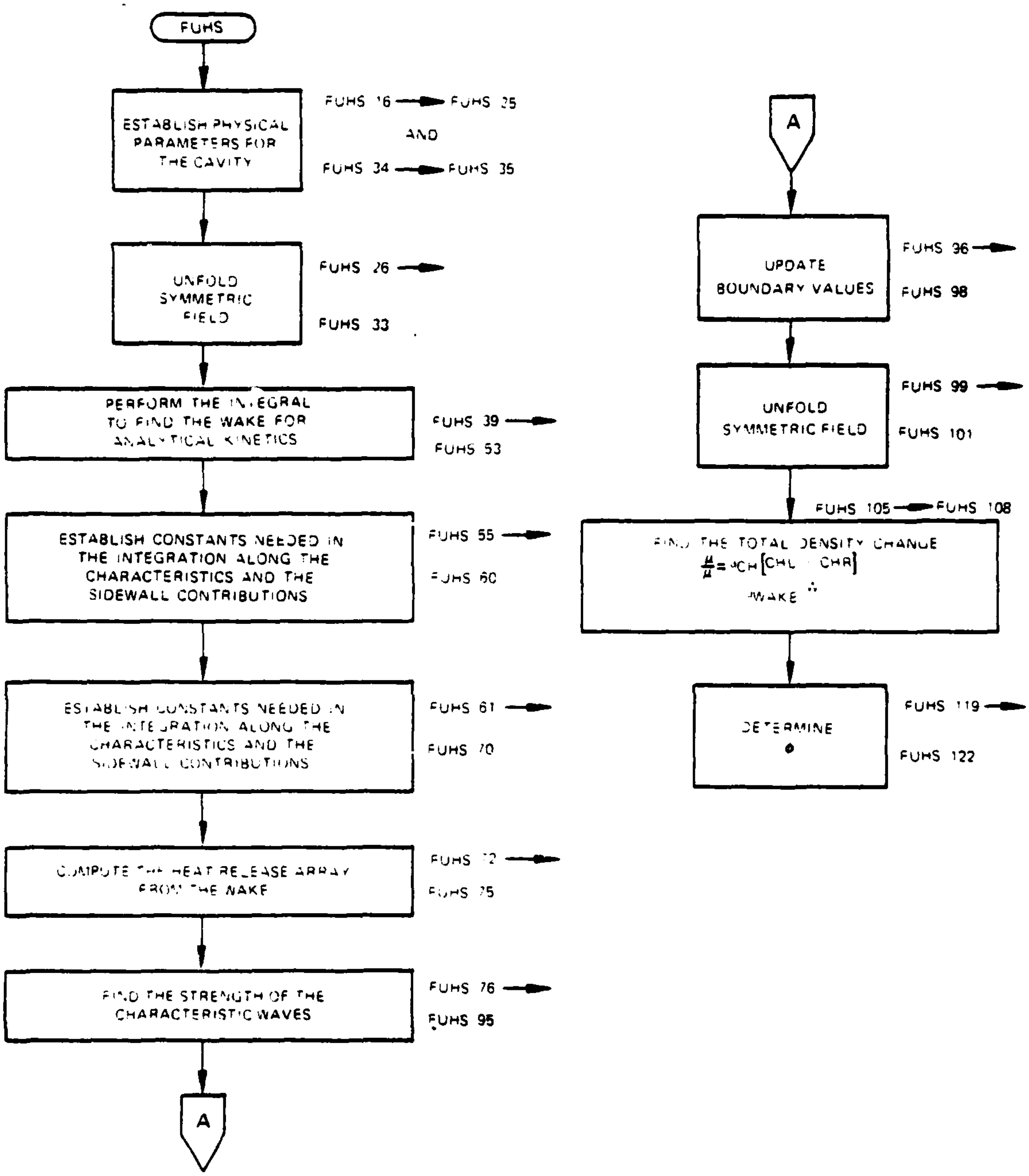


Figure 24. Subroutine FUHS organization.

Energy: $$\frac{\rho_\infty U_\infty}{\gamma - 1} \frac{\partial}{\partial} \left(\frac{P'}{P_\infty} - \frac{\gamma \rho'}{\rho_\infty} \right) = q \tag{77}$$

The solution is then found by using the potential for the flow as done by Tsien and Bieloch, (Ref. 12), resulting in the following equations for a heat source q in supersonic heat addition

$$u' = -\frac{(\gamma-1)q}{2\gamma\rho\beta}\,\delta\,(x-\beta y) \tag{78}$$

$$v' = \frac{(\gamma-1)q}{2\gamma\rho}\,\delta(x-\beta y) \tag{79}$$

$$P' = \frac{(\gamma-1)qM}{2a\beta}\,\delta\,(x-\beta y) \tag{80}$$

$$\rho' = \frac{(\gamma-1)qM}{2a^3\beta}\,\delta\,(x-\beta y) - \frac{(\gamma-1)q}{a^2 U}\,\delta\,(y)\; I\;(x) \tag{81}$$

where

$$x = \beta y \quad \text{Defines a Mach line} \tag{82}$$

$$\beta = \sqrt{M^2-1} \tag{83}$$

$$a = U/M \quad \text{Speed of sound} \tag{84}$$

$$I\;(x) = \begin{cases} 1, & x > 0 \\ 0, & x < 0 \end{cases} \tag{85}$$

For volume heat addition $q \rightarrow dq = h(x,y)dxdy$, and the effect of all sources are added; for example,

$$u' = \frac{-(\gamma-1)}{2\gamma p\beta} \iint h(x,y)\; dxdy\; \delta\; (x-\beta y) \tag{86}$$

$$= \frac{-(\gamma-1)}{2\gamma p\beta} \int_0^s h(x=\beta y)\; \sin\mu\, ds \tag{87}$$

12. Tsien, H. E. and Milton Beilock, "Heat Source in a Uniform Flow," Journal of the Aeronautical Sciences, December 1949, p. 746.

where the integral is taken along a streamline ($x = \beta y$) and $\sin\mu = 1/M$. S is related to x and y by

$$S = x \cos\mu \qquad\qquad S = y \sin\mu$$

The equation for density change is therefore,

$$\frac{\Delta\rho}{\rho} = \frac{1}{\rho}\left[\left(\frac{(\gamma-1)}{2a^3\beta}\int_0^s h(x,y)\Big|_{x=\beta y} \sin\mu ds\right)\right.$$

$$\left.-\left(\frac{(\gamma-1)}{a^2 U}\iint dx'dy' \; h(x',y')\delta\;(y-y')I(x-x')\right)\right] \tag{88}$$

The first term is due to heat addition along a streamline while the second is due to the wake in the energy release region. "Heat addition in a supersonic stream causes compression waves which radiate from the heat release region. The waves reflect from the cavity walls. Downstream of the heat release region is a wake. Whereas the compression waves increase gas density the wake decreases gas density" (Ref. 12).

The heat release (h(x,y) for a laser can be written:

$$h(x,y) = c\int_{x_{NEP}}^{x} \Delta I(x',y)\,\varepsilon^{-(x-x')/UT}\,dx' \tag{89}$$

where T is the time constant for the depopulation of the lower laser level. If the depopulation were instantaneous ($T \to 0$) then the heat release would be proportional to the intensity since for every molecule emitting a photon, that same molecule gives off a quantum of heat. It has been shown (Ref. 12) that the above equation for the heat release can be used in all regions of the far cavity with only small error.

The constant C can be found by conservation of energy. Consider the following three-level molecule shown in Figure 25.

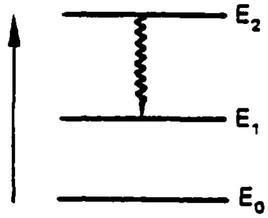


Figure 25. Three-level molecule.

The quantum efficiency η is defined as the ratio of the power out divided by the power in, so for the gain/phase segment under consideration

$$\eta = \frac{(\text{No. molecules})\ (E_2-E_1)}{(\text{No. molecules})\ (E_2-E_0)} = \frac{P}{\Delta H+\Delta P} \tag{90}$$

where

$$\Delta H = (\text{No. molecules})\ (E_1 - E_0)$$

The above expression can be inverted to give

$$\Delta H = \left(\frac{1-\eta}{\eta}\right) \Delta P$$

with

$$\Delta P = \iint dx'dy'\ \Delta I(x',y')$$

and

$$\Delta H = \iint dx'dy'\ h(x',y') \tag{91}$$

Assume, for this calculation, that (0,0) is at the corner of the sidewall and the NEP. Then,

$$\begin{aligned} \Delta H &= c\Delta z \int_0^\infty dy \int_0^\infty dx \int_0^\infty \Delta I\ (x',y)\,\varepsilon^{-(x-x')/UT} dx' \\ &= c\Delta z \int_0^\infty dy \int_0^\infty dx \int_0^\infty I\ (x-x')\ \Delta I(x',y)\,\varepsilon^{-(x-x)/UT} dx' \end{aligned} \tag{92}$$

where, recall

$$I(x-x') = \begin{cases} 1, & x>x' \\ 0 & x<x' \end{cases}$$

so

$$\Delta H = c\Delta z \int_0^\infty dy \int_0^\infty dx' \Delta I(x',y) \int_0^\infty dx I(x-x') \; \varepsilon^{-(x-x')/UT}$$

$$= c\Delta z \int_0^\infty dy \int_0^\infty dx \Delta I(x,y) \int_{x'}^\infty dx'' \varepsilon^{-x''/UT} \tag{93}$$

$$\Delta H = c\Delta z \int_0^\infty dy \int_0^\infty dx' \Delta I(x,y) \left(\frac{1}{1/UT}\right) \tag{94}$$

$$= cUT\Delta z \Delta P$$

so

$$\frac{1-\eta}{\eta} = \frac{\Delta H}{\Delta P} = cUT\Delta z \tag{95}$$

or

$$c = \left(\frac{1-\eta}{\eta}\right)\left(\frac{1}{UT\Delta z}\right) \tag{96}$$

Since the numerical kinetics return the conditions of the wake region and not the heat addition, these must be the data used. Thus, for the analytical kinetics model, find the heat addition to the wake:

$$W(x,y) = \int_0^x dx' h(x',y) = c \int_0^x dx' \int_0^{x'} dx'' \Delta I(x'',y)\, \varepsilon^{-(x'-x'')/UT}$$

$$= c \int_0^\infty dx' I(x-x') \int_0^\infty dx'' \, I(x'-x'') \, \Delta I(x'',y) \, \varepsilon^{-(x-x')/UT}$$

$$= c \int_0^\infty dx'' \Delta I(x'',y) \int_0^\infty dx' \, I(x-x') \, I(x'-x'') \, \varepsilon^{-(x'-x'')/UT}$$

$$= c \int_0^\infty dx'' \, \Delta I(x'',y) \, I(x-x'') \int_{x''}^x dx' \, \varepsilon^{-(x'-x'')/UT} \tag{97}$$

so

$$W(x,y) = c \int_0^x dx'' \, \Delta I(x'',y) \, UT\left(1-\varepsilon^{-(x-x'')/UT}\right) \tag{98}$$

so, recalling

$$c = \frac{1-\eta}{\eta} \; \frac{1}{UT\Delta z} \quad \text{and} \quad \Delta I(x',y) = 2\left(\frac{1-G}{1+G}\right) \text{PPD from SIMPGG} \tag{99}$$

wake energy addition becomes

$$W(x,y) = \frac{2}{\Delta Z} \left(\frac{1-G}{1+G}\right) \frac{1-\eta}{\eta} \int_0^x dx' \; \text{PPD} \; (x',y) \left(1-\varepsilon^{-(x-x')/UT}\right) \tag{100}$$

Now that both numerical and analytical models can give the wake integrated heat addition, the Fuhs effect is calculated in the following manner:

$$H(I,J) = \frac{1}{\Delta x} \int_{x(I-1)}^{x(I)} h(x,y)\, dx = \frac{W\left(x_{(I)}\right) - W\left(x_{(I-1)}\right)}{\Delta x} \tag{101}$$

Given this average heat release function, the integral along a characteristic can be performed. Note that reflection off the sidewalls must be included, as

can be seen in Figure 26. The contribution due to reflection at P_1 is therefore found by finding the total heat released along the characteristic that reflects at P_2, then adding this to that found along P_2P_1.

(Note: For larger Mach angles (>tan -1(1Δy/2Δx)), the effective number of points in the direction is multiplied by a factor of KS in the program so that only information in two mesh rectangles is needed to find heat addition at the wall, i.e., extrapolation from the two nearest the sidewall, as can be seen from the following more detailed description of how the left and right characteristic terms are found.) Assume KS = 1 and that the Mach angle is less than $\tan^{-1}$ (Dy/2Dx). This is assumed in the program by changing the total effective number of x coordinates to be KS*NPTS.

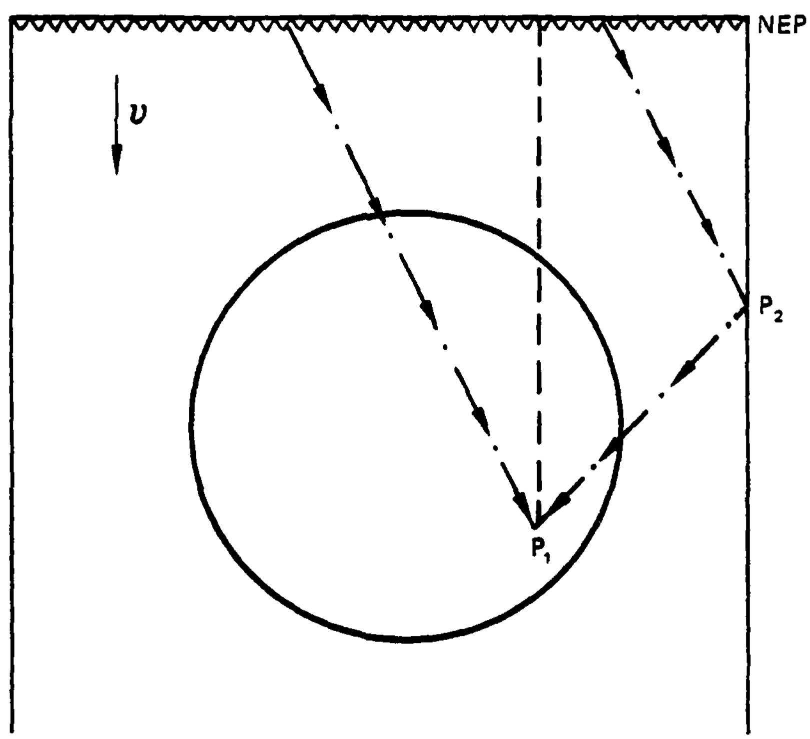


Figure 26. Average heat release function.

Consider first the left characteristic term for the (I,J) point in Figure 27:

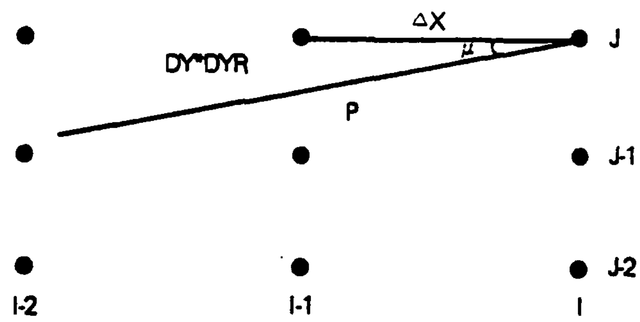


Figure 27. Left characteristic value.

The left characteristic value at (I,J) is that at P (found by a linear interpolation between the (I-1,J) and (I-1,J-1) points) plus the heat released in the region, again using a linear interpolation for H at (I-1,J) and (I-1,J-1).

Now consider a boundary point shown in Figure 28:

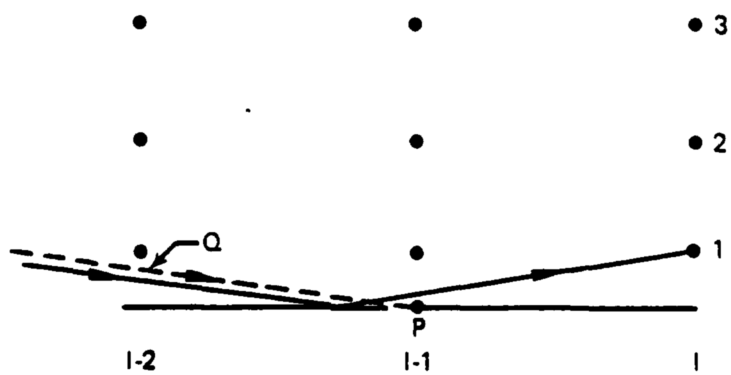


Figure 28. Boundary point.

To find the characteristic value at (I,1) it is necessary to know the value at point P which is in the (I,1) column on the sidewall. The value will then be a linear interpolation between the values at (I-1,1) and P plus a similar linear interpolation for the added heat.

To find the characteristic value at point P, the values at (I-2,2) and (I-2,1) are extrapolated linearly toward the boundary to find the value at point Q. Heat is then added, again by linear extrapolation.

Note that this detailed analysis at the boundary assumes that the characteristic of interest lies between the boundary at (I-1) and the (I-1,1) point, hence the necessity of the restriction that DYR = DYCH/DY be less than 0.5.

Analysis of the right characteristic is similar to that of the left characteristic.

The phase shift is found using the Gladstone-Dale relation.

$$n \approx 1 + C\rho \tag{102}$$

The phase change $\Delta\phi$ is

$$\Delta\phi = \frac{2\pi}{\lambda}\Delta n\Delta z = \frac{2\pi}{\lambda}\left(\frac{C}{\rho o}\Delta\rho\right)\rho_o\Delta z \tag{103}$$

This is then added to that of the unloaded density field to establish the total phase change at the gain/phase segment.

c. Fortran

Argument List

$$ZIC = \begin{Bmatrix} \text{wake for numerical kinetics} & \\ \Delta I \times \frac{1}{\Delta Z}\left(\frac{1-n}{n}\right) & \text{for analytical kinetics} \end{Bmatrix}$$

DEN = phase change returned due to the FUHS effect

NCV - cavity number

Commons Changes - none

Subroutines called - none

Computer printouts of subroutine FUHS follow.

```
      SUBROUTINE FUHS(ZIC,DEN,NCV)
C     FUHS EFFECT ALGORITHM
C     THIS ROUTINE CALCULATES THE CONTRIBUTION TO THE CAVITY DENSITY
C     FIELD DUE TO STIMULATED EMISSION INDUCED HEAT ADDITION.
      LEVEL 2, ZIC,DEN,XC
      COMMON/CAV2/XC(5),YC(5),ZC(5),NX(5),NY(5),NS(5),XMC(5),YMC(5),
     2 NGTYPE(20),              SSGAIN(190,5),SATIN(5),BETA(5),RHOS(5),
     3 VEL(5),GAM(5),XMACH(5),TV1(5),TV2(5),TV3(5),TVN2(5),TSCAV(5),
     4 PSCAV(5),PB(5),FN2(5),FCO2(5),FH2O(5),FCO(5),FO2(5),TITLE(20),
     5 AVG(5),  NSYM
      DIMENSION ZIC(  1    ),DEN(  1    ),CHR(96,2),CHL(96,2),H(96)
      ENTHP(A,B,C)=A+C*(B-A)
      CALL CPUTIM(ISRT)
C *** CALCULATE INITIAL CONSTANTS
      U = VEL(NCV)
      GMA = GAM(NCV)
      XMA = XMACH(NCV)
      RHO = RHOS(NCV)
      A = U/XMA
      AWAK = (GMA-1.0)/(A**2*U*RHO)
      IM=NX(NCV)
      JM=NY(NCV)
      DX=XC(NCV)/IM
      DY=YC(NCV)/JM
      IF (NSYM.EQ.0) GO TO 444
      J2=JM/2
      DO 445 J=1,J2
      DO 445 I=1,IM
      I2 = I +(J-1)*IM
      I3 = I + ( JM -J)*IM
      ZIC(I3)=ZIC(I2)
  445 DEN(I3)=DEN(I2)
  444 TANMU=1.0/SQRT(XMA**2-1.0)
      ACH = (GMA-1.0)*XMA/(2.0*A**3*SQRT(XMA**2-1.0)*RHO)*DY
      IF(NGTYPE(NCV) .EQ. 1)  GO TO 11
      IU=IM-1
      XLAG=DX/(U/BETA(NCV))
      DO 15 J=1,JM
      DO 14 IO=1,IU
      I=IM+1-IO
      N=I
      SUM=0.
      DO 13 IL=2,I
      N=N-1
      R = (I-N)*XLAG
      B = 0.
      IF(R.GT.20.) GO TO 12
      B = 1.0/EXP(R)
   12 CONTINUE
   13 SUM = SUM+ZIC(N+(J-1)*IM )*(1.-B)
   14 ZIC(I+(J-1)*IM)=SUM*DX
      ZIC(1+(J-1)*IM) = 0.
   15 CONTINUE
   11 DO 6 K=1,10
      KS=K
      DYCH=DX*TANMU/FLOAT(KS)
      DYR=DYCH/DY
      IF(DYR .LT. 0.5) GO TO 7
    6 CONTINUE
    7 SCL=1.+DYR
      DYR2=2.*DYR
      ACH=ACH*DYR
      SCR=1.5-DYR
      DO 1 J=1,JM
      DEN(1+(J-1)*IM) = 0.
      CHL(J,1)=0.
    1 CHR(J,1)=0.
      CHLWAL=0.
      CHRWAL=0.
```

```
      DO 200 I=2,IM                                                     FUHS
C *** COMPUTE HEAT RELEASED AT I-1                                      FUHS
      DO 210 J=1,JM                                                     FUHS
      IZ = I+(J-1)*IM                                                   FUHS
  210 H(J)=(ZIC(IZ )-ZIC(IZ-1 ))/DX                                     FUHS
C *** COMPUTE STRENGTH OF CHARACTERISTIC WAVES                          FUHS
      DO 100 K=1,KS                                                     FUHS
      DO 50 J=1,JM                                                      FUHS
C *** LEFT RUNNING WAVE                                                 FUHS
      JL=J-1                                                            FUHS
      IF(J .NE. 1) GO TO 20                                             FUHS
C *** EXTRAPOLATE FOR HEAT RELEASED, USE BOUNDARY POINT                 FUHS
      CHL(1,2)=ENTRP(CHL(1,1),CHLWAL,DYR2)+ENTRP(H(2),H(1),SCL)         FUHS
      GO TO 30                                                          FUHS
C *** INTERPOLATE FOR VALUE                                             FUHS
   20 CHL(J,2)=ENTRP(CHL(J,1),CHL(JL,1),DYR)+ENTRP(H(J),H(JL),DYR)      FUHS
C *** RIGHT RUNNING WAVE                                                FUHS
   30 JR=J+1                                                            FUHS
      IF(J .NE. JM) GO TO 40                                            FUHS
C *** EXTRAPOLATE FOR HEAT RELEASED, USE BOUNDARY POINT                 FUHS
      CHR(JM,2)=ENTRP(CHR(JM,1),CHRWAL,DYR2)+ENTRP(H(JL),H(JM),SCL)     FUHS
      GO TO 50                                                          FUHS
C *** INTERPOLATE FOR VALUE                                             FUHS
   40 CHR(J,2)=ENTRP(CHR(J,1),CHR(JR,1),DYR)+ENTRP(H(J),H(JR),DYR)      FUHS
   50 CONTINUE                                                          FUHS
C *** UPDATE BOUNDARY POINTS                                            FUHS
      CHLWAL=ENTRP(CHR(2,1),CHR(1,1),SCR)+ENTRP(H(2),H(1),SCR)          FUHS
      CHRWAL=ENTRP(CHL(JM-1,1),CHL(JM,1),SCR)+ENTRP(H(JM-1),H(JM),SCR)  FUHS
      DO 60 J=1,JM                                                      FUHS
      CHR(J,1)=CHR(J,2)                                                 FUHS
   60 CHL(J,1)=CHL(J,2)                                                 FUHS
C     WRITE(6,83) I,H(1),CHR(1,1),CHL(1,1),CHLWAL,CHRWAL                FUHS
C  83 FORMAT(1X,I5,5G12.5)                                              FUHS
  100 CONTINUE                                                          FUHS
C *** GET TOTAL DENSITY CHANGE                                          FUHS
      DO 110 J=1,JM                                                     FUHS
      IJ = I + (J-1)*IM                                                 FUHS
  110 DEN( IJ)=ACH*(CHR(J,1)+CHL(J,1))-AWAK*ZIC( IJ)                    FUHS
  200 CONTINUE                                                          FUHS
C     DO 800 K=1,8                                                      FUHS
C     WRITE(6,801)                                                      FUHS
C 801 FORMAT(1H1)                                                       FUHS
C     IL=1+16*(K-1)                                                     FUHS
C     IU=IL+15                                                          FUHS
C     DO 802 J=1,JM                                                     FUHS
C 802 WRITE(6,803) (DEN(I,J),I=IL,IU)                                   FUHS
C 803 FORMAT(1X,16(2PF6.3))                                             FUHS
C 800 CONTINUE                                                          FUHS
      ROCL = .228*RHO*ZC(NCV)/NS(NCV)                                   FUHS
      JT = IM*JM                                                        FUHS
      DO 70 J=1,JT                                                      FUHS
      DEN( J ) = ROCL*DEN( J )                                          FUHS
   70 CONTINUE                                                          FUHS
      CALL CPUTIM(IFIN)                                                 FUHS
      DELT=(ISRT-IFIN)/100.                                             FUHS
      WRITE(6,778) DELT                                                 FUHS
  778 FORMAT(20H0FUHS ANALYSIS TOOK ,G12.5,20H SECONDS OF CPU TIME,///) FUHS
      RETURN                                                            FUHS
      END                                                               FUHS
```

12. SUBROUTINE GAINXY

a. Purpose -- GAINXY controls the gain calculations in the cavity. Figure 29 shows the Subroutine GAINXY flow chart. Either small signal gain (along one stream tube) or full-field-loaded gain is selected. From input cavity conditions (including vibrational temperatures of the constituents at

nozzle exit plane), all other thermodynamic parameters, energy levels, broadened line-width function, gain, optical cross section, and saturation intensity at a single point are given. Subroutine KINET is called to integrate the rate equations along the X-direction (streamtube). This is done only once for small signal gain. When loaded gain is selected the entire field is calculated and gain is updated by local intensity one step in the Z (propagation) direction. The loaded gain is hence a numerical (small stepwise integrated) process. This updated gain and intensity field is used to SOQ.

The single stream tube small signal gain is used in subroutine SIMPGG which computes a closed form solution of the full field loaded gain.

Subroutine MIX is called by subroutine GAINXY to calculate the transition rates.

A ratio technique is employed to effect calculation of the gain field for 9.27 μlasing. This is triggered by GFACT = 1 for 10.60 μ; GFACT = 1 for 9.27 μ.

b. Relevant formalism -- The option for small signal gain only or full-field loaded numerical gain is determined by IFIELD = 1 for small signed gain and IFIELD = 1 for numerical gain.

For small signal gain only, the gain is computed first at the nozzle exit plane and then computed along the flow direction by integrating the rate equations in subroutine KINET.

The particular initial thermodynamic conditions, rotational J values (P or R branch), and initial vibrational temperatures are brought in through common/CAV2/. Then, for a particular vibration-rotation transition, the gain coefficient is given by:

$$g_{vj}^{v'j'} = \frac{8\pi^3}{3h} \left(\frac{M}{2\pi KT}\right)^{1/2} S_j F_j \left| R_{v,v'} \right|^2 \Phi \left[\frac{n_{vj}}{g_{vj}} - \frac{n_{v'j'}}{g_{v'j'}}\right] \quad (104)$$

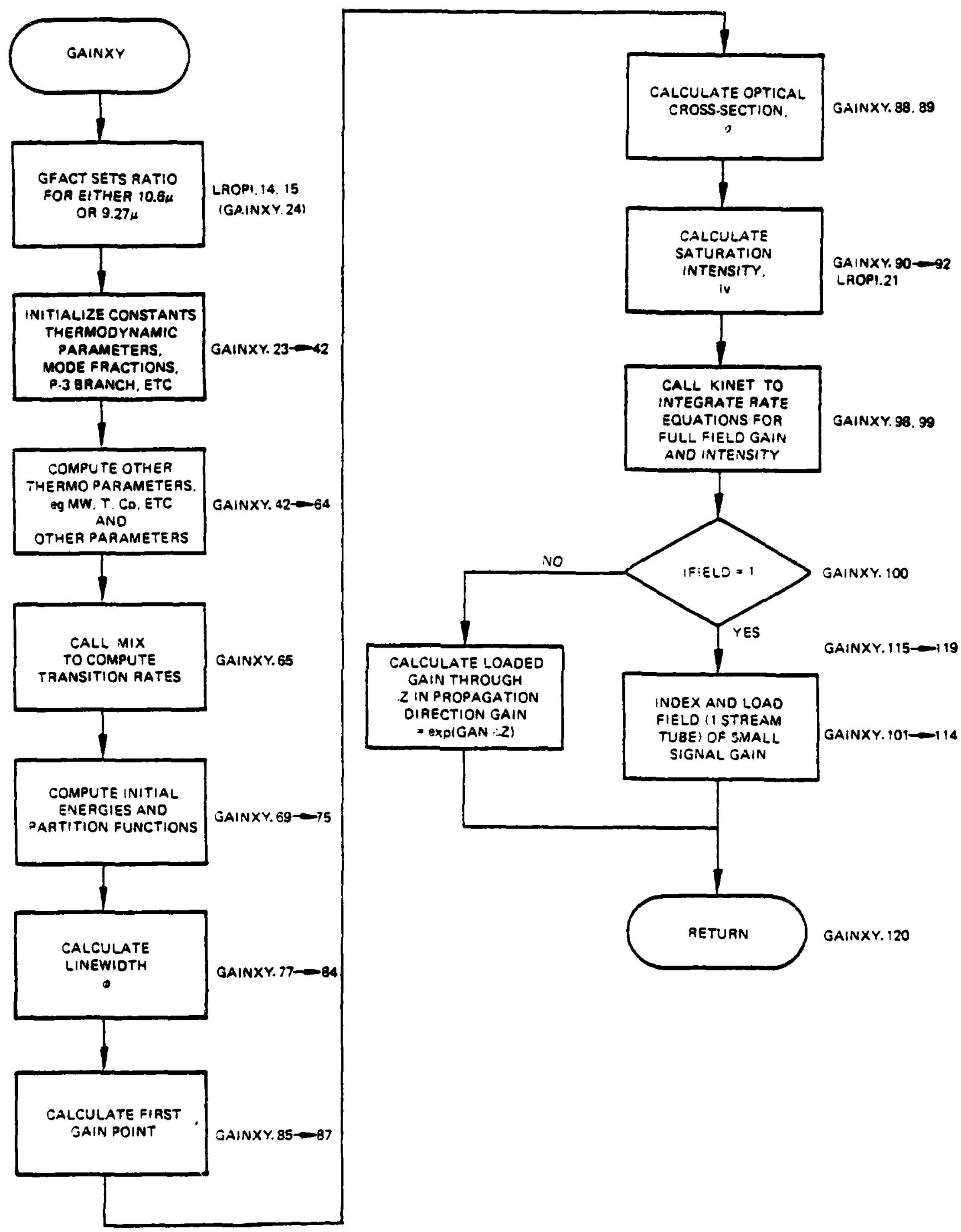


Figure 29. Subroutine GAINXY flow chart.

Where,

h = Planck's constant = 6.625×10^{-27} erg

M = Mass of CO_2 molecule = $44 \times 1.66 \times 10^{-24}$ g

K = Boltzmann's constant = 1.38×10^{-16} erg/K

$$S_J = \begin{cases} J + 1 & \text{for } J' = J + 1 \text{ (P-branch)} \\ J & \text{for } J' = J - 1 \text{ (R-branch)} \end{cases}$$

$$F_J = 1 + D_{v,v'} m \quad \text{where } M = -(J+1) \text{ P-branch}; \quad M = J \text{ R-branch}$$

$R_{vv'}$ = Vibrational matrix element for transition

ϕ = lineshape factor

$$= e^{\varepsilon^2} \operatorname{erfc}(\xi)$$

$$= (\ln 2)^{\frac{1}{2}} \frac{\alpha_p}{\alpha d}, \quad \alpha_p = \text{pressure-broadened half-width}, \quad \alpha_d = \text{Doppler-broadened half-width}$$

$$\alpha_p = \frac{n}{2\pi c} \sum_{\text{SPECIES}} X_i \bar{V}_{i-co_2} \bar{O}_{i-co_2}$$

$$\alpha_d = \frac{V_o}{C} \left(\frac{2KT \ln 2}{M} \right)^{\frac{1}{2}}$$

n = total gas number density

c = speed of light = 3×10^{10} cm/s

x_i = mole fraction of the ith species

$\bar{v}_{i-CO_2}$ = mean velocity between CO_2 and ith species

M_{i-CO_2} = reduced mass of i-CO_2 pair

α_i-CO_2 = optical broadening cross-section

Vo = frequency of transition $(v,j) - (v', j')$

$$N_{VJ} = N_V f_J = N_V \frac{2J+1}{Q^{(v)}_{rot}} \varepsilon^{\frac{-J(J+1)}{KT}} Q^{(v)}_{rot} \qquad (105)$$

where,

$$Q^{(v)}_{rot} = \frac{T}{2\Theta^{(v)}_{rot}}$$

$$\frac{N_{VJ}}{g_{VJ}} = \frac{N_V}{g_V} \frac{\exp}{Q^{(v)}_{rot}} \left(\frac{-J(J+1)\ {}^{(v)}_{rot}}{KT} \right)$$

$$\frac{N_V}{g_V} = N_{ooo} \exp\left(-\theta_V/T_V\right)$$

Θ_V = Characteristic temperature of state

T_V = Vibrational temperature of state

The saturation intensity is calculated:

$$I_{SAT} = \frac{h\nu\beta}{\sigma} \qquad (106)$$

where,

$h\nu$ = photon energy

β = lower laser level relaxation rate

σ = optical cross-section of the transition

Where Rc2 is the EOVO transition rate ~ (1/s), all the initial energies of the vibration levels are computed before entering subroutine KINET.

$$EOOVI = \frac{X_{co_2} * 2349}{\varepsilon^{\frac{hc*2349}{KT_2}} - 1}$$

$$EOVOI = \frac{X_{co_2}*2349}{\varepsilon^{\frac{hc*667}{KT_2}} - 1}$$

$$EVOOI = \frac{X_{co_2}*2349}{\varepsilon^{\frac{hc*1388}{KT_1}} - 1}$$

$$EN2I = \frac{X_{N2}*2331}{\frac{hc*2331}{KTX_{N2}} - 1} \qquad (107)$$

Where X_{co_2} and X_{N2} are mole fractions of CO_2 and N_2, and T1, T2, T_{N2} are vibrational temperatures. These vibrational temperatures and levels are shown schematically in Figure 30.

Gain is computed as a function of x by calling "KINET."

When the loaded numerical gain option is triggered (IFIELD) ≠ 1), the full field (in X and Y) gain is calculated in KINET as a function of previous intensities and the field is updated when returned to GAINXY by propagating each local intensity through a ΔZ, with local gain GAN(I). The gain is thus recomputed for each point $G(J) = e^{G(J) \cdot \Delta Z}$.

Argument List

XIC	intensity array of propagation field
GAN	gain array of propagation field
NCV	cavity indicator
IFIELD	trigger for small signal gain (= 1) for full field loaded gain (≠ 1)

Commons Modified

/START/

TSI	static temperature (K)
PSI	static pressure (atm)
VI	gas velocity (cm/s)

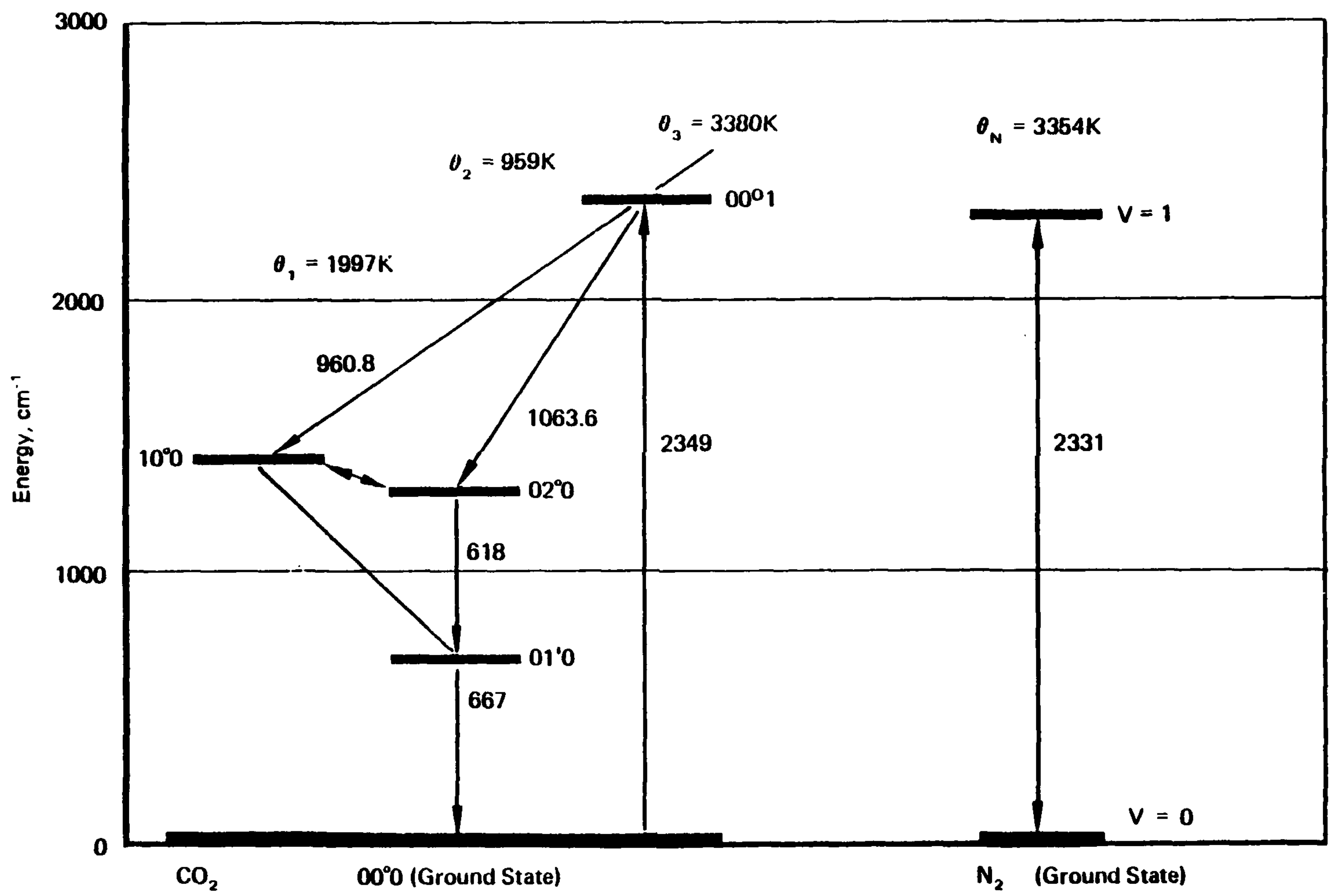


Figure 30. Characteristic temperature and energy levels.

EOOVI	Initial Energy (OOV level)
EOVOI	Initial Energy (OVO level)
EN2I	Initial Energy N_2 vibrational level
GAINI	INITIAL GAIN

/PROPT/

TS	static temperature (K)
PS	static pressure (atm)
V	gas velocity (cm/s)
RHO	gas density (g/cm^2)
RHON	number density (cm^{-3})
CP	specific heat @ constant pressure
GAMMA	ratio of specific heats
R	gas constant of mixture
B	(ln 2) (3.78×10^6)
XLAMB	wavelength (λ)
HNU	energy of photon of wavelength XLAMB
CPRM	parameter to get Doppler broadened line width ratio

/MOLES/

XN2	mole fraction (N_2)
XCO2	mole fraction (CO_2)
XH2O	mole fraction (H_2O)
XCO	mole fraction (CO)
XO2	mole fraction (O_2)

/RATE/

RSTIM	stimulated transition rate (s^{-1})

/FACTOR/

XMW	molecular weight of gas mixture

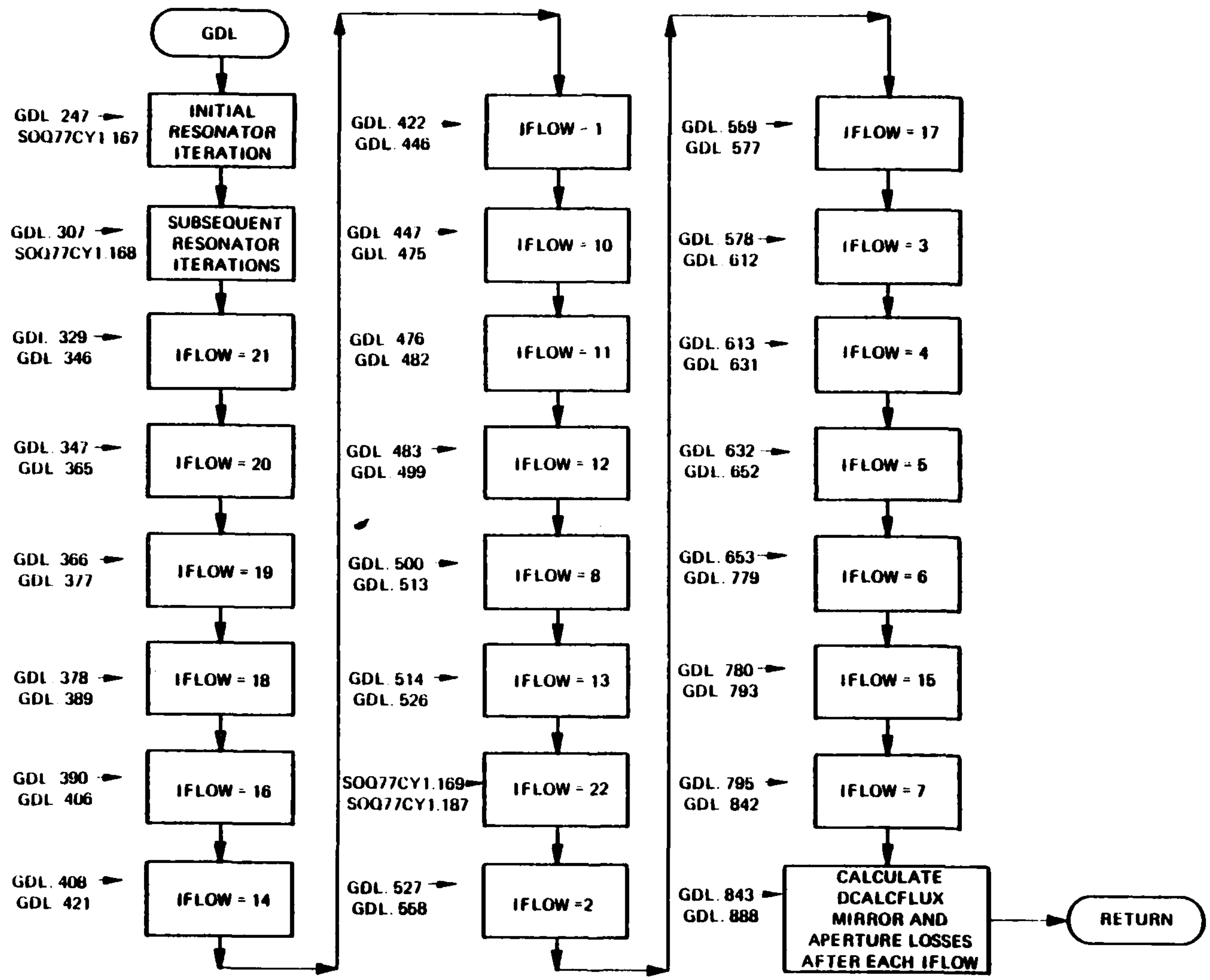


Figure 31. Subroutine GDL organization.

AG	Avogadro's number
GCON	gain correction factor
ROTUP	upper rotational level (K)
ROTLO	lower rotational level (K)
RCORR	correction factor for optical x-section
C	speed of light (cm/s)

SUBROUTINE GAINXY   76/176   OPT=1   FIN 4.6+452   04/27/79   12.23.47

```
      SUBROUTINE GAINXY(XIC,GAN,NCV,IFIELD)                             GAINXY      2
C     NUMERICAL GAIN ROUTINE                                            GAINXY      3
C     THIS ROUTINE CALCULATES:1. SMALL SIGNAL GAIN FOR USE IN SIMPGG    GAINXY      4
C                             2. NUMERICAL LOADED GAIN                  GAINXY      5
C            ******************                                         GAINXY      6
C     IFIELD = 1 FOR SMALL SIGNAL GAIN ONLY                             GAINXY      7
C            ******************                                         GAINXY      8
      LEVEL 2, XIC,GAN,XC                                               GAINXY      9
      COMMON/START/TSI,PSI,VI,E0UVI,EUV0I,EV00I,EN2I,GAINI              GAINXY     10
      COMMON /GFACTR/ GFACT(2)                                          LROP1      13
      COMMON/PROPT/TS,PS,V,RHO,RHON,CP,GAMMA,R,B,XLAMB,RNU,CPRM         GAINXY     11
      COMMON/MOLES/XN2,XCO2,XH2O,XCO,XO2                                GAINXY     12
      COMMON/ENERG/EN2,E0UV,EUV0,EV00                                   GAINXY     13
      COMMON/RATE/RN2,RC3,RC2,RPUMP,      RSTIM                         GAINXY     14
      COMMON/FACTER/XM*,AG,GCON,ROTUP,ROTLO,RCORR,C                     GAINXY     15
      COMMON/CAV2/XC(5),YC(5),ZC(5),NX(5),NY(5),NS(5),XMC(5),YMC(5),    GAINXY     16
     2 NGTYP(20),  SSGAIN(190,5),SATIN(5),BETA(5),RMOS(5),              GAINXY     17
     3 VEL(5),GAM(5),XMACH(5),TV1(5),TV2(5),TV3(5),TVN2(5),TSCAV(5),    GAINXY     18
     4 PSCAV(5),PBCH(5),FN2(5),FCO2(5),FH2O(5),FCO(5),FO2(5),           GAINXY     19
     5 TITLE(20),AVG(5),NSYM                                            GAINXY     20
      DIMENSION XIC(  1  ) ,GAN(  1  )                                  GAINXY     21
      CALL CPUTIM(ISRT)                                                 GAINXY     22
      TSI=TSCAV(NCV)                                                    GAINXY     23
      WLFACT=1.                                                         LROP1      14
      IF (GFACT.NE.1.) WLFACT = 10.6/9.27                               LROP1      15
C * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *   GAINXY     24
      SLOPIN = 2.15                                                     GAINXY     25
C * * * * ** ** *** **  ** ** ** ** **** *** ** ** ** ** ** ** ** ** ** GAINXY     26
      PSI=PSCAV(NCV)                                                    GAINXY     27
      VI=VEL(NCV)                                                       GAINXY     28
      PB=PBCH(NCV)                                                      GAINXY     29
      XN2=FN2(NCV)                                                      GAINXY     30
      XCO2=FCO2(NCV)                                                    GAINXY     31
      XH2O=FH2O(NCV)                                                    GAINXY     32
      XCO=FCO(NCV)                                                      GAINXY     33
      XO2=FO2(NCV)                                                      GAINXY     34
      T1 =TV1(NCV)                                                      GAINXY     35
      T2 =TV2(NCV)                                                      GAINXY     36
      T3 =TV3(NCV)                                                      GAINXY     37
      TN2=TVN2(NCV)                                                     GAINXY     38
      TS = TSI                                                          GAINXY     39
      PS = PSI                                                          GAINXY     40
      V = VI                                                            GAINXY     41
      R0 = 8.317E7                                                      GAINXY     42
C GFACT MODIFIES GAIN                                                   LROP1      16
      GCON = .991E-14*PB*GFACT(NCV)                                     LROP1      17
      ROTUP = (PB-1.)*PB*.556                                           GAINXY     44
      ROTLO = PB*(PB+1.)*.561                                           GAINXY     45
      AG = 6.023E23                                                     GAINXY     46
      XN2 = XN2+XCO                                                     GAINXY     47
```

```
      XMW = 28.016*XN2+44.011*XCO2+18.016*XH2O+32.0*XO2                 GAINXY    48
      XO2FAC=20.939                                                     LHOP1     18
      IF (GFACT.NE.1.) XO2FAC=18.526                                    LHOP1     19
      RCORR = .669* XN2      /SQRT(17.12/22.005) + XCO2 + .292*XH2O/    GAINXY    49
     X SQRT(12.783/22.005)+.646*XO2/SQRT(XO2FAC/22.005)                 LHOP1     20
      SIGMA = 13.E-15                                                   GAINXY    51
      RCORR = SIGMA*RCORR*.165E-7                                       GAINXY    52
      C = 3.E10                                                         GAINXY    53
      B = .69315*0.0378E8                                               GAINXY    54
      H = 6.625E-27                                                     GAINXY    55
      XLAMB = 1.439/(1380.+(PB-1.)*PB*.556-PB*(PB+1.)*.561)             GAINXY    56
      HNU = H*C/XLAMB                                                   GAINXY    57
      CPRM = RCORR*C*XLAMB/SQRT(B)                                      GAINXY    58
      R = R0/XMW                                                        GAINXY    59
      GAMMA = (7.*(XN2+XCO2+XO2)+8.*XH2O)/(5.*(XN2+XCO2+XO2)+6.*XH2O)   GAINXY    60
      RHO = PS/R/TS*1.013E6                                             GAINXY    61
      GAM(NCV)=GAMMA                                                    GAINXY    62
      XMACH(NCV)=V1/SQRT(GAMMA*R*TS1)                                   GAINXY    63
      RHOS(NCV)=RHO                                                     GAINXY    64
      CALL MIX                                                          GAINXY    65
      BETA(NCV)=RC2                                                     GAINXY    66
      RHON = RHO/XMW*AG                                                 GAINXY    67
      CP = 3.5*R0          *(XN2+XCO2+XO2+8./7.*XH2O)                   GAINXY    68
      E00VI = XCO2*2349./(EXP(1.439*2349./T3)-1.)                       GAINXY    69
      E0V0I=  XCO2*1334./(EXP(1.439*667./T2)-1.)                        GAINXY    70
      EV00I = XCO2*1388./(EXP(1.439*1388./T1)-1.)                       GAINXY    71
      EN2I = XN2*2331./(EXP(1.439*2331./TN2)-1.)                        GAINXY    72
      Q1 = 1./(1.-EXP(-1997./T1))                                       GAINXY    73
      Q2 = 1./(1.-EXP(-960./T2))**2                                     GAINXY    74
      Q3 = 1./(1.-EXP(-3380./T3))                                       GAINXY    75
      X000 = XCO2/(Q1*Q2*Q3)                                            GAINXY    76
C     CALCULATE LINEWIDTH                                               GAINXY    77
      APAD = CPRM*RHON                                                  GAINXY    78
      WORM = .8326*APAD                                                 GAINXY    79
      IF(WORM.GT.10.) GO TO 40                                          GAINXY    80
      PHI = EXP(WORM**2)*ERFC(WORM)                                     GAINXY    81
      GO TO 41                                                          GAINXY    82
   40 PHI = 0.67764/APAD                                                GAINXY    83
   41 CONTINUE                                                          GAINXY    84
      TFACT = TS**(-1.5)                                                GAINXY    85
      GAIN = GCON*TFACT*RHON*X000*PHI*(.556*EXP(-3380./T3-ROTUP/TS)     GAINXY    86
     X  -.561*EXP(-1997./T1-ROTLO/TS))                                  GAINXY    87
C     OPTICAL CROSS SECTION                                             GAINXY    88
      BIGSIG = GCON*TFACT*PHI*EXP(-ROTUP/TS)*.556                       GAINXY    89
C     SATURATION INTENSITY                                              GAINXY    90
      SATIN(NCV)=HNU*RC2/BIGSIG/1.E7                                    GAINXY    91
      IF (NGTYP(NCV).EQ.0)     SATIN(NCV)=SATIN(NCV) * SLOPIN           GAINXY    92
      SATIN(NCV) = SATIN(NCV) * WLFACT                                  LHOP1     21
      RSTIM = 0.0                                                       GAINXY    93
      GAINI = GAIN                                                      GAINXY    94
      IXMAX=NX(NCV)                                                     GAINXY    95
      IY=NY(NCV)/(NSYM+1)                                               GAINXY    96
      DXCAV=XC(NCV)/IXMAX                                               GAINXY    97
C     CALCUALTE GAIN AS A FUNCTION OF X                                 GAINXY    98
      CALL KINET(XIC,GAN,IXMAX,DXCAV,IFIELD,IY)                         GAINXY    99
      IF(IFIELD .NE. 1) GO TO 980                                       GAINXY   100
C     INITIALIZE SMALL SIGNAL GAIN                                      GAINXY   101
      DO 300 I = 1,IXMAX                                                GAINXY   102
  300 SSGAIN(I,NCV)=GAN(I)                                              GAINXY   103
      SATINK=SATIN(NCV)/1000.                                           GAINXY   104
      WRITE(6,100) GAM(NCV),XMACH(NCV),RHOS(NCV),BETA(NCV),SATINK       GAINXY   105
  100 FORMAT(2/H0RESULTS FROM KINETICS DECK/1X,8HGAMMA = ,G12.5,4X,15HMA GAINXY   106
     XCH  NUMBER = ,G12.5,4X,10HDENSITY = ,G12.5,4X,7HBETA = ,G12.5,    GAINXY   107
     X 4X,8HSATIN = ,G12.5///27X,*(18H  XNEP     G0(XNEP)))             GAINXY   108

      DO 101 I=1,IXMAX                                                  GAINXY   109
      GAN(I+IXMAX)=100.*SSGAIN(I,NCV)                                   GAINXY   110
  101 GAN( I )=(2*I-1)*DXCAV/2.                                         GAINXY   111
      WRITE(6,102) (GAN( I ),GAN(I+IXMAX) ,I=1,IXMAX)                   GAINXY   112
      GO TO 982                                                         GAINXY   113
  102 FORMAT(40(25X,8F9.3/))                                            GAINXY   114
```

```
C       CALCULATE LOADED GAIN                                           GAINXY     115
  980 DELTAZ=ZC(NCV)/NS(NCV)/2.                                         GAINXY     116
      MUT = IXMAX*IY                                                    GAINXY     117
      DO 981 J=1,MUT                                                    GAINXY     118
  981 GAN( J )=EXP(GAN( J )*DELTAZ)                                     GAINXY     119
  982 RETURN                                                            GAINXY     120
      END                                                               GAINXY     121
```

13. SUBROUTINE GDL

a. Purpose -- Subroutine GDL is the main driver program for resonator and optical train calculations. It is here that the information about each resonator element is stored, as well as the order in which they are applied to the beam. Figure 31 shows the Subroutine GDL organization.

b. Formalism -- Subroutine GDL controls the iterative procedure of starting with a given field established in the main program (SOQ) and propagates this field through the resonator. Eventually, the mode which loses the least power (in the case of a bare resonator) or gains the most power (in the case of a loaded resonator) will predominate since the other modes will be suppressed due to relative power loss. For the degenerate case when two or more modes are competing for the status of lowest loss mode, the field will usually fail to converge to a single mode shape, since there is no unique mode for that eigenvalue.

c. Fortran -- To accomplish the above, GDL contains several fundamental arrays. One is the singly dimensioned CU array in which the field is stored. For a given point (x(I), x(J)) the field value is stored in the complex location.

CU (I + (J-1) * NPTS)

Common /MELT/ contains CU as well as the work array CFIL, the coordinate array x, the location of the optical axis (DRX and DRY), and the iteration number NITER. This common is shared by most of the routines in the deck. The other major arrays are the ABC array, the IGDL array, and the GNOT array. During the first iteration of a particular run, GDL reads input from unit IN in the form of namelists and titles. The order of resonator elements to be met by the beam is controlled by the order in which the $CONTRL cards are read. These contain the IFLOW parameters which designate specific elements, as follows:

NAMELIST/CONTROL/IFLOW, SNOTE, IPLOTS

IFLOW CONTROLS THE FLOW OF CALCULATIONS THROUGH GDL

= 1 CAVITY ELEMENT, READS CAVTY1, CAVTY2. (from CAVITY)

= 2 MIRROR ELEMENT, READS MIROR

= 3 VAMP ELEMENT, READS PROPGT

= 4 APERTURE ELEMENT, READS APTUR

= 5 THERMAL BLOOMING, READS BLOOM

= 6 INTERPOLATE FIELD OVER SMALLER AREA, READS CUTOUT

= 7 TEST FOR CONVERGENCE OF ITERATION, NO INPUT

= 8 PLOT FIELD DISTRIBUTION, READS TITLE

= 9 RETURN CONTROL TO CALLING PROGRAM, NO INPUT

= 10 READ AND/OR WRITE CU ON DISK, READS DISKIT

= 11 AERO WINDOW R.M.S. PHASE MODEL, NO INPUT

= 12 SCALING ROUTINE . . .MULTIPLIES ENTIRE FIELD, READS MULT

= 13 FLIPS THE FIELD ABOUT THE y-AXIS, NO INPUT

= 14 SINUSOIDAL DENSITY VARIATIONS, READS SINDEN

= 15 REGRIDS FIELD TO LARGER SIZE, READS REGRID

= 16 CU PUNCHED ON CARDS, NO INPUT

= 17 MIRROR THERMAL BL MODEL, READS THRML

= 18 SPIDER ROUTINE, READS SPIDR

= 19 AXION ROUTINE, READS AXICON

= 20 PROPAGATE IN R-THETA SPACE, READS RPROP

= 21 REMOVES OR ADDS BACK BEAM CENTER, READS CENTER

= 22 FLIPS THE BEAM ABOUT THE x-AXIS, NO INPUT

IPLOTS is the printer plot selector. IPLOTS=ABCDE, where A=1 selects R-theta plots, B=1 selects iso-intensity plot, C=1 selects x-axis plot, D=1 selects diagonal plot, and E=1 selects y-axis plot: example, IPLOTS = 1001 selects

iso-intensity and y-axis plots in x-y coordinates. The order of IFLOW numbers for a given resonator is then stored in the IGDL array for future iterations. In the same manner the associated titles are stored in the GNOT array.

Usually for a given IFLOW there is another associated namelist containing relevant element parameters. Once read in, these numbers are stored in ABC (I,J,K) where I indicates the parameter for the J the element of type K. The number (J) of the element is stored in common ZIP, which is equivalenced to the ICAVZ array. At the beginning of each iteration most of ICAVZ is filled with zeros so that the center index of the ABC array is correctly identified. At the end of each iteration, the current field is compared with that of the previous iteration in two ways: (1) the cutout and interpolated feedback field is compared and (2) the full field just before the hole-coupling mirror is compared. When the differences between two consecutive iterations fall within given tolerances (10% for the feedback field, 2% for the hole-coupler field and 0.7% for the power at the output of the resonator), the field is said to have converged, i.e., the lowest loss mode has been selected. A more detailed description of the meaning of each IFLOW, its function, and its associated namelist, if any, follows:

IFLOW = 1 (GDL. 422→GDL.446)

A GDL cavity is applied to the field. NEWCAV is calculated to see if the beam has been in the cavity before. The namelist used in CAVTY1.

CALLS CAVITY.

NAMELIST/CAVTY1/NCAVNO, ILR, NSTE, NPLT, ZPROP1, ZPROPO

NCAVNO IS THE NUMBER ASSIGNED TO CAVITY FOR IDENTIFICATION

ILR INDICATES DIRECTION OF FIELD THROUGH CAVITY

= -1 RIGHT TO LEFT

= +1 LEFT TO RIGHT

NSTE CONTROLS TYPE OF VAMP CODE BETWEEN SEGMENTS

= 1 CONSTANT MESH WITH SETUP

= 2 VARIABLE MESH WITH SETUP (EXITS VAMP AT END OF ELEMENT)

= 3 VARIABLE MESH WITH SETUP (REMAINS IN VAMP)

= 4 USE EXISTING PROPAGATING MATRIX (EXITS VAMP)

= 5 USE EXISTING PROPAGATING MATRIX (REMAINS IN VAMP)

NPLT CONTROLS INTERMEDIATE PRINTOUT FOR CAVITY

= 0 NO PRINTOUT

= 1 PRINT FIELD BEFORE AND AFTER GAIN, AND GAIN COEFFICIENT

ZPROPI IS PROPAGATION DISTANCE FROM PREVIOUS OPTICAL ELEMENT TO CAVITY.

ZPROPO IS PROPAGATION DISTANCE FROM CAVITY TO NEXT OPTICAL ELEMENT

IFLOW = 2 (GDL.527→GDL.558)

Here the parameters necessary for application of a mirror are set up. The namelist read is MIROR. CALLS MIRROR

NAMELIST/MIROR/ANGXX, ANGYY, RADC, DIAOUT, DIAIN, XMPOS, YMPOS, RMIR, X DELIA, DISTF, DDUTY, DINY, RANULS, PHIAST

ANGXX IS TILT IN x-DIRECTION - RADIANS (WRT OPT. AXIS)

ANGYY IS TILT IN y-DIRECTION - RADIANS (WRT OPT. AXIS)

RADC IS RADIUS OF CURVATURE OF SPHERICAL MIRROR

DIAOUT IS OUTSIDE DIAMETER OF MIRROR

DIAIN IS INSIDE DIAMETER OF MIRROR

XMPOS IS X-DISPLACEMENT OF MIRROR FROM OPTICAL AXIS

YMPOS IS Y-DISPLACEMENT OF MIRROR FROM OPTICAL AXIS

RMIR IS REFLECTIVITY OF MIRROR

DELTA IS CENTER-TO-EDGE DISTORTION FACTOR (CM)

DISTF IS MIRROR DISTORTION FACTOR (DEFLECTION=DISTF*I*(1.0-RMIR))

RANULS IS OUTSIDE RADIUS OF ANNULAR BEAM (IF APPLICABLE)

DOUTY FLAGS THE TYPE OF APERTURE APPLIED -

.EQ. 0 - CIRCULAR APERTURE DEFINED AS ABOVE

.NE. 0 - RECTANGULAR APERTURE, DIAOUT HIGH (X) BY DOUTY WIDE (Y)

DINY IS SIMILAR TO DDUTY FOR INSIDE DIMENSIONS

PHIAST IS THE ANGLE OF INCIDENCE OF THE BEAM IN DEGREES

IFLOW = 3 (GDL.578→GDL.612)

For this IFLOW, a propagation step is applied. Relevant parameters are found in namelist PROPGT. CALLS STEP.

NAMELIST/PROPGT/DELZ, RDCURV, WINDOX, WINDOK, IIFG, IITR, IIPS

DELZ IS PROPAGATION DISTANCE

RDCURV IS RADIUS OF CURVATURE OF PHASE FRONT

IF (ABS (RDCURV) .LT.0.5) USE RADCUR OF PREVIOUS

MIRROR

WINDOX IS X-SPACE DATA WINDOW FOR FFT

WINDOK IS K-SPACE DATA WINDOW FOR FFT

IIFG IS A VAMP CONTROL PARAMETER

= 1 FOR CONSTANT MESH

= 2 FOR VARIABLE MESH

IITR IS ANOTHER VAMP CONTROL PARAMETER

= 0 NO INVERSE TRANSFORM

= 1 INVERSE TRANSFORM BACK TO REAL SPACE

IIPS IS FOR CORRECTION OF PLANE AND SPHERICAL PHASE

FRONTS

= 0 NO CORRECTION

= 1 PLANAR CORRECTION ONLY

= 2 QUADRATIC CORRECTION ONLY (NOT OPERATIONAL)

= 3 BOTH

IFLOW = 4 (GDL.613→GDL.631)

Here an aperture is applied. IF DOUT and DIN are both less than 0, SLIVER is called. If both are greater than or equal to zero, APRTR is called. The relevant namelist is APTUR.

AD-A103 285 UNITED TECHNOLOGIES CORP WEST PALM BEACH FLA F/G 20/5
SYSTEM OPTICAL QUALITY USERS GUIDE. PART 2.(U)
MAR 80 J L FORGHAM, S S TOWNSEND F29601-77-C-0025

UNCLASSIFIED AFWL-TR-79-141-PT-2 NL

2 OF 3
AD A
103285

NAMELIST/APTUR/DOUT, DIN, XPOS, YPOS, YOUT, YIN

DOUT IS OUTSIDE DIAMETER OF APERTURE

DIN IS INSIDE DIAMETER OF APERTURE

XPOS IS x-DISPLACEMENT OF APERTURE FROM OPTICAL AXIS
YPOS IS y-DISPLACEMENT OF APERTURE FROM OPTICAL AXIS

YOUT FLAGS THE TYPE OF APERTURE APPLIED -

.EQ.0 - CIRCULAR APERTURE DEFINED AS ABOVE

.NE.0 - RECTANGULAR APERTURE, DOUT HIGH (X) BY
YOUT WIDE (Y)

YIN IS SIMILAR TO YOUT FOR INSIDE DIMENSIONS

IFLOW = 5 (GDL.632→GDL.652)

Thermal Blooming is applied to the complex field. BLOOM is read in and subroutine TBLOOM is called.

NAMELIST/BLOOM/ALFA, SCP, T, RHO, ZLEN, NSTEPS, INPT, NPROP, AXIAL, DT

AFLA	=	MEDIUM ABSORPTION COEFFICIENT, CM^{-1}
SCP	=	MEDIUM SPECIFIC HEAT, J/GM-DEG K
T	=	MEDIUM TEMPERATURE, DEG K
RHO	=	MEDIUM DENSITY, GM/CM3 (OR TRANSVERSE VEL. IF .GT.1.)
ZLEN	=	MEDIUM THICKNESS ALONG OPTICAL AXIS
NPROP	=	PROPAGATION PARAMETER. . .SAME AS NSTE IN CAVITY
NSTEPS	=	NUMBER OF ELEMENTS IN SUBSYSTEM, .GE. 1
INPT	=	.NE.0 FOR INTERMEDIATE FIELD PLOTS
AXIAL	=	AXIAL VELOCITY (CM/SEC) IF .GT. 0, USES AXIAL BLOOMING
DT	=	BEAM ON TIME FOR THERMAL BOUNDARY LAYER GROWTH IN TRANSIENT BLOOMING CALCS. IF DT.GT.0 USES TRANSIENT BLOOMING

IFLOW = 6 (GDL.653→GDL.779)

For this option the field can be cut out and interpolated from one region size to another. The number of points is not changed. If CUSMF is not equal to zero, the field-averaged feedback field is stored on unit 8 and the convergence checks are made on the feedback field and the pre-HCM field which is stored on unit 7 temporarily. The field for the bare-resonator is renormalized at this point to unit maximum intensity. Namelist CUTOUT has the information for the new region in it as well as other parameters.

NAMELIST/CUTOUT/DIBEAM, OVRLAP, DXXR, DYYR, MAXIT, AVCUSM, CUSMF

CUSMF = 1. FOR NORMAL LOADED RESONATOR CUTOUT

CUSMF = 0. AVOIDS WRITING FIELD ON 8 AND AVOIDS NORMALIZING FIELD, CHANGES TO THE NEW COORDINATES, THEN RETURNS.

DIBEAM IS THE DIAMETER OF BEAM FOR NEXT ITERATION

OVRLAP IS DCALC = OVRLAP*DIBEAM

DXXR IS POSITION OF ITERATIVE BEAM REL. TO OPTICAL AXIS

DYYR IS THE SAME

MAXIT IS THE MAXIMUM NUMBER OF ITERATIONS

AVCUSM AVERAGES PREVIOUS AND NEXT ITERATION GUESS IN THE

HOPE OF RAPID CONVERGENCE...=0 NO AVE, = .5 HALF AND HALF

IFLOW = 7 (GDL.795→GDL.842)

There is no namelist associated with this option. The convergence check on the power is made here. If the solution has not yet converged, the gain/phase information is updated by a call to REGAIN, then the resonator is restarted for the next pass.

IFLOW = 8 (GDL.500→GDL.513)

If the parameter plot is non-zero in namelist START in SOQ, this IFLOW will generate printer plots by a call to IPLOT. Namelist PLOT is read.

NAMELIST/PLOT/TITLE RADPLT

TITLE IDENTIFIES THE POSITION OF EACH STATION PLOTTED

RADPLT CONTROLS THE TYPE OF PLOT

= 0.0 FOR X,Y PLOTTING (X-AXIS, Y-AXIS, DIAGONAL)

= 1.0 FOR RADIAL PLOTTING AT VARIOUS THETAS

IFLOW = 9

This IFLOW only results in the return to the main program, SOQ.

IFLOW = 10 (GDL.447→GDL.475)

This option allows the field to be read in from or read to a specific unit in standard SOQ format. It calls no peripheral subroutines and reads the unit designation from namelist DISKIT.

NAMELIST/DISKIT/IREAD, IWRITE, IORD, IADD

IREAD IS THE DISK # TO BE READ OFF/ON... IF=0...DON'T READ

IWRITE IS THE DISK # TO BE WRITTEN ON... =0...DON'T WRITE

IORD IS THE ORDER = 1, READ FIRST

=-1, WRITE FIRST

IADD = 1 UPDATES IWRITE BY 1 FOR SUCCESSIVE ITERATIONS

IFLOW = 11 (GDL.476→GDL.482)

This option applies an aerodynamic window to the complex field. It reads no namelist and calls AEROW to perform the calculation.

IFLOW = 12 (GDL.483→GDL.499)

The field can be scaled using this option. At the same time the x array can also be magnified. No subroutines are called and MULT is read.

NAMELIST/MULT/TRANS, XMAG

TRANS IS TRANSMISSION OF ELEMENT

XMAG IS MAGNIFICATION FACTOR FOR THE X-ARRAY

IFLOW = 13 (GDL.514→GDL.526)

This option flips the field about its y-axis. No namelists are read and no subroutine called.

IFLOW = 14 (GDL.408→GDL.421)

This option imposes a sinusoidal density (phase) variation to the existing complex field. It calls no subroutines, but it reads SINDEN for information on the sine wave.

NAMELIST/SINDEN/NBEAM, AWL

NBEAM IS THE NUMBER OF CYCLES PER X-CALCULATED REGION

AWL IS THE AMP/WL OF THE SINUSOIDAL VARIATIONS

IFLOW = 15 (GDL.780→GDL.793)

The field can have superimposed on it a different number of mesh points. The spacing between two adjacent points does not change unless RGRD is called. Just the number of points in the mesh changes. If the number of points is increased, RGRD adds zeros to the outside of the existing region. This option reads namelist REGRID

NAMELIST/REGRID/NGRD

NGRD IS NO. OF FIELD POINTS ACROSS REGRIDDED DCAL

IFLOW = 16 (GDL.390→GDL.406)

In this IFLOW, no subroutine is called and no input is read. The field and coordinates are written format to TAPE 4 in cards to be punched.

IFLOW = 17 (GDL.559→GDL.557)

Quiescent thermal gradients are imposed by this option. Namelist THRML is read and subroutine THERML is called.

NAMELIST/THRML/ALPHAM, CONMIR, ALPHAG, RHOGAS, TAU, TIN, REFMIR, CONGAS

THRML IS THE NAMELIST FOR BOUNDARY LAYER THERMAL LENS CALCULATIONS

ALPHAM = MIRROR DIFFUSIVITY (CM2/SEC)

CONMIR = MIRROR THERMAL CONDUCTIVITY (WATTS/CM-SEC)

ALPHAG = THERMAL DIFFUSIVITY OF GAS HEATED BY MIRROR (CM2/SEC)

CONGAS = THERMAL CONDUCTIVITY OF GAS HEATED BY MIRROR (WATT/CM-SEC)

RHOGAS = DENSITY OF GAS HEATED BY MIRROR (GM/CC)

TAU = BEAM ON TIME FOR BOUNDARY LAYER GROWTH (SEC)

TIN = INITIAL TEMPERATURE OF GAS & MIRROR (DEG K)

REFMIR = MIRROR REFLECTIVITY (OBTAINED FROM MIRROR INPUT)

THERMAL MAY BE APPLIED AFTER ANY MIRROR TO ALTER THE GAIN-PHASE DUE TO HEATING OF THE QUIESCENT BOUNDARY LAYER ADJACENT TO THE MIRROR SURFACE.

IFLOW = 18 (GDL.378→GDL.389)

With IFLOW = 18, a spider obscuration can be applied. Subroutine SPIDER is called using the information read in with namelist SPIDR.

NAMELIST/SPIDR/NSPD, WIDTH, THETA, XSPC, YSPC, DIH

NSPD = NUMBER OF STRUTS IN SPIDER (MAX=6)

WIDTH = WIDTH OF SPOKES IN SPIDER

THETA = ANGLE OF INDIVIDUAL SPOKES OF SPIDER

XSPC = x-LOCATION OF CENTER OF SPIDER

YSPC = y-LOCATION OF CENTER OF SPIDER

DIH = HUB DIAMETER

IFLOW = 19 (GDL.366→GDL.377)

This option allows for the application of an axicon. Subroutine AXICN is called after namelist AXICON is read.

NAMELIST/AXICON/CAPR, EXPAND, ROC, DISP, TILT

CAPR IS THE OUTSIDE RADIUS OF THE ANNULAR EXTRACTION BEAM.

(EXPAND.EQ. .TRUE.) MEANS THE BEAM IS GOING FROM CIRCULAR TO ANNULAR IN CROSS-SECTION

ROC = RADIUS OF CURVATURE OF THE FIELD IN PHYSICAL SPACE

DISP = DISPLACEMENT OF AXICON FROM CENTER ALONG X-AXIS

TILT = ANGLE (RADIANS) OF AXICON TILT FROM DIRECTION OF PROP.

IFLOW = 20 (GDL.347→GDL.365)

This option propagates an unrolled annulus. After reading in namelist RPROP, it then calls subroutines RSTEP to perform the propagation and POWR to determine the power after propagation.

NAMELIST/RPROP/DELZR, DELZTH, WINDOX, WINDOK

DELZTH IS PROPAGATION DISTANCE FOR THE RADIAL COORDINATE

DELZTH IS PROPAGATION DISTANCE FOR THE ANGULAR COORDINATE

(DELZR .NE. DELZTH) MEANS YOU ARE MAKING AN EQUIVALENT COLLIMATED BEAM PROPAGATION STEP IN R-THETA COORDINATES

WINDOX IS X-SPACE DATA WINDOW FOR FIT

WINDOK IS K-SPACE DATA WINDOW FOR FFT

IFLOW = 21 (GDL.329→GDL.346)

This option allows for the removal of the center of the beam which is then stored on unit 20, or it can allow for the addition of a field read from unit 20 modified by a phase change. This work is all done in subroutine FIELDS using the information read in from namelist CENTER

NAMELIST/CENTER/DSM, REMOVE, PHIARB

DSM IS THE DIAMETER TO BE REMOVED AND LATER ADDED TO THE MAIN BEAM

REMOVE FLAGS THE ACTION

.TRUE. IF THE CENTER PORTION OF THE BEAM IS TO BE REMOVED

.FALSE. IF THE REMOVED PORTION IS TO BE ADDED BACK TO THE BEAM

PHIARB IS AN ARBITRARY PHASE CHANGE ADDED TO THE CENTRAL PORTION

IFLOW = 22   (SOQ77CY1.169→SOQ77CY1.181)

This option flips the field about the x-axis. No input is required and no subroutines are called.

Argument List

IN - INPUT UNIT FOR RESONATOR DATA

RESTRT - NEW OR OLD RESONATOR?

ABC - PARAMETER ARRAY

NITER - CURRENT ITERATION

IB - INPUT UNIT # OF OLD FIELD

IFLAG -= 1 TRANSFERS TO OLD ENTRY POINT - READS FIELD FROM IB/ CONTINUES.

ABC and NITER can be redefined by this subroutine.

Common Variables Modified:

The common variable not modified by GDL are:

WL, NPTS, NPY, RADCUR, WNOW and NREG. Note that NBC is modified by its equivalence with IGDL and IDIR.

SUBROUTINE GDL   76/176   OPT=1   FIN 4.6+452   04/27/79   12.23.47

```
      SUBROUTINE GDL(IN,RESTRT,ABC,NITER,IB,IFLAG)                      GDL
C     OPTICAL CALCULATIONS ROUTING ROUTINE                              GDL
C     BY MEANS OF THE INPUT (&CONTRL)THE USER INSTRUCTS THIS ROUTINE    GDL
C     TO DIRECT THE CALCULATION OF OPTICAL EFFECTS OF APERTURES,        GDL
C     MIRRORS, CAVITIES, ETC.                                           GDL
C  IFLAG=1 TRANSFERS TO OLD AUTO ENTRY POINT                            GDL
C*****************************************************************C GDL
C     IN IS UNIT CONTAINING INPUT DATA FOR CONFIGURATION                GDL
C     RESTRT IS CONTROL FOR RESTARTING CALCULATIONS FROM PREVIOUS RUN   GDL
C           = .TRUE. IF RESTARTING                                      GDL
C           = .FALSE. IF NOT                                            GDL
C*****************************************************************C GDL
C                                                                       GDL
      LEVEL 2, CU,CFIL2,CFFL                                            GDL
      COMMON/MELT/CU(16384),CFIL(16512),A(128),WL,NPTS,NPY,DRX,DRY      GDL
      COMMON/MRPROP/RADCUR,ANGX,ANGY                                    GDL
      COMMON/ WAY / WNOW,NREG,RAPTH                                     GDL
      COMMON/ZIP/ICAV,IMIR,ISTEP,NOS,IAP,IPTT,ITRAN,ITRRML,IAX,IRSIP,   GDL
```

```
 X ICUT,MLT,IDR,ITM,ICEK,NCT                                            GOL        20
  COMMON /QAZ/ APLT(30,20), NBC(180),  SAVE(10)                         GOL        21
  COMMON /INITL/ INT                                                    GOL        22
  DIMENSION IDIR(4,24),IGOL(99),ABC(12,20,9),CFFL(16384),IDSK(4,9),     C10FLA      1
 XTAY(262),XK(128),XDUM(128),ERROR(10),TITLE(20),CFIL2(16384),Y(128)    GOL        24
 X,ZLI(12),ZLO(12),GNOTE(20),GNOT(50,20),THETA(6),CPR(4),XPND(4),       C10FLA      2
 X  DSMM(20),RMV(20),PHIA(20),RCURVE(4),DSP(4), TLT(4),ICAVZ(16)        GOL        26
  DIMENSION IPLTS(50)                                                   GOL        27
  COMPLEX CFFL,CFIL2,CU,CFIL,CPCNT,CFACTT,CUD                           GOL        28
  LOGICAL INIT,RESTRT,WHY    ,  EXPAND,XPND , REMOVE,RMV                GOL        29
  EQUIVALENCE (NBC(1),IGOL(1)),(NBC(100),IDIR(1,1))                     C10FLA      3
  EQUIVALENCE (CFIL(1),CFFL(1)),(CFIL2(1),CU(1)), (ICAV,ICAVZ(1))       GOL        32
  DATA IFLOW,TITLE /9,20*4H     /  , IPLOTS / 0 /                       GOL        33
  DATA NCAVNO,ILR,NSTE,NPLT,ZPROPI,ZPROPO /0,1,1,0,2*0./                GOL        34
  DATA ANGXX,ANGYY,RADC,DIAOUT,DIAIN,XMPOS,YMPOS,RMIR,DELTA,DISTF       GOL        35
 X    /0.0,    0.0,  0.0, 0.0,    0.0,  0.0,    0.0, 1.0, 0.0 , 0.0/    GOL        36
  DATA RANULS, DOUTY, DINY, PHIAST                                      C10ASTG     2
 X /0.0, 0.0, 0.0, 0.0/                                                 C10ASTG     3
  DATA DELZ, RDCURV, WINDOX,WINDOK,IIFG,IITR,IIPS                       GOL        39
 X      /0.0,    0.0,    0.1,    0.1,   1,   0,    0/                   GOL        40
  DATA DOUT,DIN,XPOS,YPOS,YOUT,YIN/ 6*0.0/                              SQAPR      16
  DATA DIBEAM,OVRLAP, DXXR, DYYR, MAXIT, AVCUSM /4*0.0,1,0.0/           GOL        42
  DATA CUSMF/1./                                                        CYCLE9      2
  DATA RADPLT/0.0/                                                      GOL        43
  DATA ALFA,SCP,T,RHO,ZLEN,NSTEPS,INPT,NPROP,AXIAL/5*0.0,1,1,0,0.0/     GOL        44
  DATA DT /0.0/                                                         GOL        45
  DATA IREAD, IWRITE, IORD, IADD  /0,0,1,0/                             GOL        46
  DATA TRANS, XMAG /1.0,1.0/                                            GOL        47
  DATA NBEAM,AWL /0,0.0/                                                GOL        48
  DATA NGRD /2/                                                         GOL        49
  DATA ALPHAM,CONMIR,ALPHAG,RHOGAS,TAU,TIN,REFMIR,CONGAS                GOL        50
 X/6*0.0,1.0,0.0/                                                       GOL        51
  DATA CAPR,EXPAND,ROC /30.,.TRUE.,0.0/, DISP,TILT/0.,0./               GOL        52
  DATA DELZR, DELZTH, WINDOX, WINDOK                                    GOL        53
 X      /0.0,    0.0,     0.1,     0.1/                                 GOL        54
  DATA DSM , REMOVE , PHIARB                                            GOL        55
 X      /0.0, .TRUE. ,  0.0  /                                          GOL        56
  DATA DIM,XSPC,YSPC,WIDTH,THETA,NSPD/10.14,0.,0.,.423,-120.,           CORR2       6
 X 0.,4*0.,2/                                                           CORR2       7
  NAMELIST/ CONTRL / IFLOW,GNOTE,IPLOTS                                 GOL        57
C                                                                       GOL        58
C   IFLOW CONTROLS THE FLOW OF CALCULATIONS THROUGH GOL                 GOL        59
C         = 1 CAVITY ELEMENT, READS CAVTY1,CAVTY2                       GOL        60
C         = 2 MIRROR ELEMENT, READS MIROR                               GOL        61
C         = 3 VAMP ELEMENT, READS PROPGT                                GOL        62
C         = 4 APERTURE ELEMENT, READS APTUR                             GOL        63
C         = 5 THERMAL BLOOMING, READS BLOOM                             GOL        64
C         = 6 INTERPOLATE FIELD OVER SMALLER AREA, READS CUTOUT         GOL        65
C         = 7 TEST FOR CONVERGENCE OF ITERATION, NO INPUT               GOL        66
C         = 8 PLOT FIELD DISTRIBUTION, READS TITLE                      GOL        67
C         = 9 RETURN CONTROL TO CALLING PROGRAM, NO INPUT               GOL        68
C         = 10  READ AND/OR WRITE CU ON DISK, READS DISKIT              GOL        69
C         = 11  AERO WINDOW R.M.S. PHASE MODEL, NO INPUT                GOL        70
C         = 12  SCALING ROUTINE...MULTIPLIES ENTIRE FIELD, READS MULT   GOL        71
C         = 13  FLIPS THE FIELD ABOUT THE Y-AXIS, NO INPUT              GOL        72
C         = 14  SINUSOIDAL DENSITY VARIATIONS, READS SINDEN             GOL        73
C         = 15  REGRIDS FIELD TO LARGER SIZE, READS REGRID              GOL        74
C         = 16  CU PUNCHED ON CARDS, NO INPUT                           GOL        75
C         = 17  MIRROR THERMAL BL MODEL, READS THRML                    GOL        76
C         = 18  SPIDER ROUTINE, READS SPIDR                             GOL        77
C         = 19  AXICN ROUTINE, READS AXICON                             GOL        78
C         = 20  PROPAGATE IN R-THETA SPACE, READS RPROP                 GOL        79
C         = 21  REMOVES OR ADDS BACK BEAM CENTER, READS CENTER          GOL        80
C          = 22  FLIPS THE BEAM ABOUT THE X-AXIS, NO INPUT              SOU77CY1  165
C                                                                       GOL        81
C   IPLOTS IS THE PRINTER PLOT SELECTOR. IPLOTS=ABCDE WHERE A=1 SELECTS GOL        82
C   R-THETA PLOTS, B=1 SELECTS ISO INTENSITY PLOT , C=1 SELECTS X AXIS  GOL        83
C   PLOT, D=1 SELECTS DIAGONAL PLOT, AND E=1 SELECTS Y AXIS PLOT,.      GOL        84
C   EXAMPLE---IPLOTS=1001 SELECTS ISO INTENSITY AND Y AXIS PLOTS IN     GOL        85
C   X-Y COORDINATES.                                                    GOL        86
   NAMELIST /CAVTY1/ NCAVNO,ILR,NSTE,NPLT,ZPROPI,ZPROPO                  GOL        87
```

```
C                                                                       GOL       88
C     NCAVNO IS THE NUMBER ASSIGNED TO CAVITY FOR IDENTIFICATION        GOL       89
C     ILR INDICATES DIRECTION OF FIELD THROUGH CAVITY                   GOL       90
C          = -1 RIGHT TO LEFT                                           GOL       91
C          = +1 LEFT TO RIGHT                                           GOL       92
C     NSTE CONTROLS TYPE OF VAMP CODE BETWEEN SEGMENTS                  GOL       93
C          = 1 CONSTANT MESH WITH SETUP                                 GOL       94
C          = 2 VARIABLE MESH WITH SETUP (EXITS VAMP AT END OF ELEMENT)  GOL       95
C          = 3 VARIABLE MESH WITH SETUP (REMAINS IN VAMP)               GOL       96
C          = 4 USE EXISTING PROPAGATING MATRIX (EXITS VAMP)             GOL       97
C          = 5 USE EXISTING PROPAGATING MATRIX (REMAINS IN VAMP)        GOL       98
C     NPLT CONTROLS INTERMEDIATE PRINTOUT FOR CAVITY                    GOL       99
C          = 0 NO PRINTOUT                                              GOL      100
C          = 1 PRINT FIELD BEFORE AND AFTER GAIN, AND GAIN CO-EFF       GOL      101
C     ZPROPI IS PROPAGATION DISTANCE FROM PREVIOUS OPT. ELEMENT TO CAV. GOL      102
C     ZPROPO IS PROPAGATION DISTANCE FROM CAV. TO NEXT OPTICAL ELEMENT  GOL      103
C                                                                       GOL      104
      NAMELIST/MIROR/ANGXX,ANGYY,RADC,DIAOUT,DIAIN,XMPOS,YMPOS,RMIR,    GOL      105
     X DELTA,DISTF,DOUTY,DINY,RANULS,PHIAST                             C10ASTG    4
C                                                                       GOL      107
C     ANGXX IS TILT IN X-DIRECTION - RADIANS (WRT OPT. AXIS)            GOL      108
C     ANGYY IS TILT IN Y-DIRECTION - RADIANS (WRT OPT. AXIS)            GOL      109
C     RADC IS RADIUS OF CURVATURE OF SPHERICAL MIRROR                   GOL      110
C     DIAOUT IS OUTSIDE DIAMETER OF MIRROR                              GOL      111
C     DIAIN  IS INSIDE  DIAMETER OF MIRROR                              GOL      112
C     XMPOS IS X-DISPLACEMENT OF MIRROR FROM OPTICAL AXIS               GOL      113
C     YMPOS IS Y-DISPLACEMENT OF MIRROR FROM OPTICAL AXIS               GOL      114
C     RMIR IS REFLECTIVITY OF MIRROR                                    GOL      115
C     DELTA IS CENTER-TO-EDGE DISTORTION FACTOR (CM)                    GOL      116
C     DISTF IS MIRROR DIST. FACTOR (DEFLECTION=DISTF*I*(1.0-RMIR))      GOL      117
C     RANULS IS OUTSIDE RADIUS OF ANNULAR BEAM (IF APPLICABLE)          GOL      118
C     DOUTY FLAGS THE TYPE OF APERTURE APPLIED -                        SQAPR     18
C            .EQ. 0 - CIRCULAR APERTURE DEFINED AS ABOVE                SQAPR     19
C            .NE. 0 - RECTANGULAR APERTURE, DIAOUT HIGH (X) BY DOUTY WIDE SQAPR   20
C     DINY IS SIMILAR TO DOUTY FOR INSIDE DIMENSIONS                    SQAPR     21
C    PHIAST IS THE ANGLE OF INCIDENCE OF THE BEAM --- DEGREES           C10ASTG    5
C                                                                       GOL      119
      NAMELIST/ PROPGT / DELZ, RDCURV,WINDOX,WINDOK,IIFG,IITR,IIPS      GOL      120
C                                                                       GOL      121
C     DELZ IS PROPAGATION DISTANCE                                      GOL      122
C     RDCURV IS RADIUS OF CURVATURE OF PHASE FRONT                      GOL      123
C          IF ABS(RDCURV) LT 0.5 USE RADCUR OF PREVIOUS MIRROR          GOL      124
C     WINDOX IS X-SPACE DATA WINDOW FOR FFT                             GOL      125
C     WINDOK IS K-SPACE DATA WINDOW FOR FFT                             GOL      126
C     IIFG IS A VAMP CONTROL PARAMETER                                  GOL      127
C          = 1 FOR CONSTANT MESH                                        GOL      128
C          = 2 FOR VARIABLE MESH                                        GOL      129
C     IITR IS ANOTHER VAMP CONTROL PARAMETER                            GOL      130
C          = 0  NO INVERSE TRANFORM                                     GOL      131
C          = 1  INVERSE TRANSFORM BACK TO REAL SPACE                    GOL      132
C                                                                       GOL      133
C     IIPS IS FOR CORRECTION OF PLANE AND SPHEREICAL PHASE FRONTS       GOL      134
C          = 0  NO CORRECTION                                           GOL      135
C          = 1  PLANAR CORRECTION ONLY                                  GOL      136
C          = 2  QUADRATIC CORRECTION ONLY (NOT OPERATIONAL)             GOL      137
C          = 3  BOTH                                                    GOL      138
C                                                                       GOL      139
      NAMELIST /APTUR/ DOUT,DIN,XPOS,YPOS,YOUT,YIN                      SQAPR     22
C                                                                       GOL      141
C     DOUT IS OUTSIDE DIAMETER OF APERTURE                              GOL      142
C     DIN  IS INSIDE  DIAMETER OF APERTURE                              GOL      143
C     XPOS IS X-DISPLACEMENT OF APERTURE FROM OPTICAL AXIS              GOL      144
C     YPOS IS Y-DISPLACEMENT OF APERTURE FROM OPTICAL AXIS              GOL      145
C     YOUT FLAGS THE TYPE OF APERTURE APPLIED -                         SQAPR     23
C            .EQ. 0 - CIRCULAR APERTURE DEFINED AS ABOVE                SQAPR     24
C            .NE. 0 - RECTANGULAR APERTURE, DOUT HIGH (X) BY YOUT WIDE (Y SQAPR   25
C     YIN IS SIMILAR TO YOUT FOR INSIDE DIMENSIONS                      SQAPR     26
C                                                                       GOL      146
      NAMELIST /CUTOUT/ DIBEAM,OVRLAP,DXXR,DYYR,MAXIT,AVCUSM,CUSMF      CYCLE9     3
C  CUSMF=1. FOR NORMAL LOADED RESONATOR CUTOUT                          CYCLE9     4
C  CUSMF=0. AVOIDS WRITING FIELD ON 8 AND AVOIDS NORM. FIELD UNLOADED   CYCLE9     5
```

```
C
C     DIBEAM IS THE DIAMETER OF BEAM FOR NEXT ITERATION
C     OVRLAP IS:   DCALC= OVRLAP*DIBEAM
C     DXXR IS  POSITION OF ITERATIVE BEAM REL. TO OPTICAL AXIS
C     DYYR IS  THE SAME
C     MAXIT IS THE MAX NUMBER OF ITERATIONS
C
C             AVCUSM AVERAGES PREVIOUS NEXT ITERATION GUESS IN THE HOPE
C             OF RAIPID CONVERGENCE... = 0 NO AV,  =.5  ITS HALF AND HALF
C
      NAMELIST/ PLOT /  TITLE , RADPLT
C           TITLE IDENTIFIES THE POSITION OF EACH STATION PLOTTED
C           RADPLT CONTROLS THE TYPE OF PLOT
C                = 0.0 FOR X,Y PLOTTING (X-AXIS, Y-AXIS, DIAGONAL)
C                = 1.0 FOR RADIAL PLOTTING AT VARIOUS THETAS
C
      NAMELIST/ BLOOM / ALFA,SCP,T,RHO,ZLEN,NSTEPS,INPT,NPROP,AXIAL,DT
C
C         ALFA  = MEDIUM ABSORBTION COEFFICIENT, CM-1
C         SCP   = MEDIUM SPECIFIC HEAT,J/GM-DEG K
C         T     = MEDIUM TEMPERTURE, DEG K
C         RHO   = MEDIUM DENSITY, GM/CM3 (OR TRANSVERSE VEL. IF .GT. 1.)
C         ZLEN  = MEDIUM THICKNESS ALONG OPTICAL AXIS
C         NPROP = PROPAGGGATION PARAMETER...SAME AS NSTE IN CAVITY
C         NSTEPS = NUMBER OF ELEMENTS IN SUBSYSTEM, .GE. 1
C         INPT  = .NE. 0 FOR INTERMEDIATE FIELD PLOTS
C         AXIAL = AXIAL VELOCITY (CM/SEC) IF GT 0, USES AXIAL BLOOMING
C         DT    = BEAM ON TIME FOR THERMAL BDY LAYER GROWTH IN TRANSIENT
C                 BLOOMING CALCS. IF DT GT 0. USES TRANSIENT BLOOMING
C
      NAMELIST/ DISKIT / IREAD, IWRITE, IORD , IADD
C         IREAD IS THE DISK NUM TO BE READ OFF OF...IF=0...DON'T READ
C         IWRITE IS THE DISK # TO BE WROTE ON          =0...DON'T WRITE
C         IORD IS THE ORDER   = 1, READ FIRST
C                            =-1, WRITE FIRST
C         IADD = 1 UPDATES IWRITE BY 1 FOR SUCCESSIVE ITERATIONS
C
      NAMELIST / MULT /  TRANS, AMAG
C          TRANS IS TRANSMISSION OF ELEMENT
C
      NAMELIST / SINDEN /  NBEAM,  AWL
C        NBEAM IS THE NUMBER OF CYCLES PER X-CALCULATED REGION
C        AWL IS THE AMP/WL OF THE SINUMSOIDAL VARIATIONS
C
      NAMELIST /REGRID/ NGRD
C        NGRD IS NO. OF FIELD POINTS ACROSS REGRIDDED DCAL
C
C
      NAMELIST /THRML/ALPHAM,CONMIR,ALPHAG,RHOGAS,TAU,TIN,REFMIR,
     XCONGAS
C
C   THRML IS THE NAMELIST FOR BOUNDARY LAYER THERMAL LENS CALCULATIONS
C     ALPHAM= MIRROR DIFFUSIVITY            CMSQ/SEC
C     CONMIR= MIRROR THERMAL CONDUCTIVITY                WATTS/CM SEC
C     ALPHAG= THERMAL DIFFUSIVITY OF GAS HEATED BY MIRROR CMSQ/SEC
C     CONGAS= THERMAL CONDUCTIVITY OF GAS HEATED BY MIRROR WATT/CM-SEC
C     RHOGAS= DENSITY OF GAS HEATED BY MIRROR                GM/CC
C     TAU    = BEAM ON TIME FOR BOUNDARY LAYER GROWTH         SEC
C     TIN    = INITIAL TEMPERATURE OF GAS & MIRROR          DEG K
C     REFMIR= MIRROR REFLECTIVITY (OBTAINED FROM MIRROR INPUT)
C &THERML&MAY BE APPLIED AFTER ANY MIRROR TO ALTER THE GAIN - PHASE
C  DUE TO HEATING OF THE QUIESCENT BOUNDARY LAYER ADJACENT TO THE
C  MIRROR SURFACE.
C
      NAMELIST/ SPIDR /NSPD,WIDTH,THETA,XSPC,YSPC,DIH
C       NSPD   = NUMBER OF STRUTS IN SPIDER (MAX=6)
C       WIDTH = WIDTH OF SPOKES IN SPIDER
C       THETA = ANGLE OF INDIVIDUAL SPOKES OF SPIDER
C       XSPC   = X-LOCATION OF CENTER OF SPIDER
C       YSPC   = Y-LOCATION OF CENTER OF SPIDER
C       DIH    = HUB DIAMETER
C
```

```
      NAMELIST / AXICON /  CAPR,EXPAND,ROC,DISP,TILT
C          CAPR IS THE OUTSIDE RADIUS OF THE ANNULAR EXTRACTION BEAM
C        EXPAND EQ. .TRUE. MEANS THE BEAM IS GOING FROM CIRCULAR TO
C              ANNULAR IN CROSS-SECTION
C        ROC = RADIUS OF CURVATURE OF THE FIELD IN PHYSICAL SPACE
C        DISP = DISPLACEMENT OF AXICON FROM CENTER ALONG X-AXIS
C        TILT = ANGLE(RADIANS) OF AXICON TILT FROM DIRECTION OF PROP.
C
      NAMELIST/ RPROP / DELZR,DELZTH,WINDOX,WINDOK
C
C     DELZR IS PROPAGATION DISTANCE FOR THE RADIAL COORDINATE
C     DELZTH IS PROPAGATION DISTANCE FOR THE ANGULAR COORDINATE
C        ***  DELZR .NE. DELZTH MEANS YOU ARE MAKING AN EQUIVALENT   ***
C        ***  COLLIMATED BEAM PROPAGATION STEP IN R-THETA COORDINATES **
C     WINDOX IS X-SPACE DATA WINDOW FOR FFT
C     WINDOK IS K-SPACE DATA WINDOW FOR FFT
C
      NAMELIST/ CENTER / DSM,REMOVE,PHIARB
C
C     DSM IS THE DIAMETER TO BE REMOVED AND LATER ADDED TO THE MAINBEAM
C     REMOVE FLAGS THE ACTION -
C        .TRUE. IF THE CENTER PORTION OF THE BEAM IS TO BE REMOVED
C        .FALSE. IF THE REMOVED PORTION IS TO BE ADDED BACK TO THE BEAM
C     PHIARB IS AN ARBITRARY PHASE CHANGE ADDED TO THE CENTRAL PORTION
C***********************************************************************C
C
      IF (IFLAG.NE.0) GO TO 4752
      CALL CPUTIM(ISTRT)
      IGNAL=1
      RAPTR=0.0
      SPPW=1.E70
      CPCNT=0.0
      MSTEP=0
      WHY  = .TRUE.
      KAUTO = 0
      NIT = NITER
      ICNTL=0
      ANGX=0.
      ANGY=0.
C     CALL ZERO(ICAV,NCT)
      DO 173 IZERO=1,16
  173 ICAVZ(IZERO)=0
C     CALL ZERO(GNOT(1,1),GNOT(50,20))
      DO 174 IZERO=1,20
      DO 174 JZERO=1,50
  174 GNOT(JZERO,IZERO) = 0.
      DO 3 I6=1,10
    3 SAVE(I6)=1.
      NOB = NPTS*NPY
C     ******************************************************************
C     BEGIN DIRECTION OF OPTICAL CALCULATIONS
C1000 CALL ZERO(GNOTE(1),GNOTE(20))
 1000 DO 176 IZERO=1,20
  176 GNOTE(IZERO)=0.
      READ(IN,CONTRL)
      IGATE = 0
      READ (IN,1243) GNOTE
 1243 FORMAT (20A4)
      ICNTL=ICNTL+1
      IPLTS(ICNTL) = IPLOTS
      DO 802 I=1,20
      GNOT(ICNTL,I)=GNOTE(I)
  802 CONTINUE
      WRITE(6,801) (GNOT(ICNTL,I),I=1,20)
  801 FORMAT(//1X,30(3H***)/5X,20A4/1X,30(3H***))
C     CALL CPUTIM(INCR)
C     TIME=(ISTRT-INCR)/100.
C     ISTRT=INCR
C     IF(NITER.EQ.0.0)WRITE(6,1002)TIME
C1002 FORMAT(//20X,27HCPU TIME SINCE LAST CONTRL=,F8.2//)
      ITM = ITM+1
      INIT = .TRUE.
      IOOL(ITM) = IFLOW
```

```
C  IFLOW =   /1/ 2/ 3/ 4/ 5/ 6/ 7/ 8/  9/ 10/ 11/ 12/ 13/ 14/ 15/
      GO TO (10,20,30,40,50,60,70,80,900,100,340,350,360,420,150
C      /16 /17 / 18/ 19/ 20/ 21/
     X,160,170,180,190,200,210,365),IFLOW
C     ENTRY AUTO(ABC,IB)
 4752 READ (IB) (CU(IZ),IZ=1,NOB),X,DWW,WWW
      REWIND IB
      KAUTO = 1
      NIT = 0
      DHX = ABC(1,2,1)
      DHY = ABC(2,2,1)
      NITER = 0
      WHY = .TRUE.
C     ***********************************************************
C     RESTART POINT FOR SECOND AND SUBSEQUENT ITERATIONS OF A RESONATOR
   99 NCT = 0
      ICNTL=0
      INIT=.FALSE.
      ANGX=0.
      ANGY=0.
C     CALL ZERO(ICAV,IDK)
      DO 177 IZERO=1,13
  177 ICAVZ(IZERO) = 0
      IWHA = 0
   98 IWHA = IWHA + 1
      ICNTL=ICNTL+1
      IPLOTS = IPLTS(ICNTL)
      IGATE = 0
      IFLOW=IGOL(IWHA)
      WRITE(6,801)(GNOT(ICNTL,I),I=1,20)
C  IFLOW =   /1/ 2/ 3/ 4/ 5/ 6/ 7/ 8/  9/ 10/ 11/ 12/ 13/ 14/ 15/
      GO TO (10,20,30,40,50,60,70,80,900,100,340,350,360,420,150
C      /16/17/ 18/ 19/ 20/ 21/
     X,160,160,180,190,200,210,365),IFLOW
      STOP
C     ***********************************************************
C     CUT OUT FIELD CENTER AND SAVE OR ADD TO CURRENT FIELD
  210 ICUT = ICUT+1
      IF(.NOT. INIT) GO TO 212
      READ(IN,CENTER)
      DSMM(ICUT) = DSM/2.
      RMV(ICUT) = REMOVE
      PHIA(ICUT) = PHIARB
  212 IF(.NOT.RMV(ICUT)) GO TO 216
      WRITE (6,214)  DSMM(ICUT)
  214 FORMAT(/29H  THE BEAM CENTER ( RADIUS = ,F6.3,20H ) HAS BEEN REMOV
     XED //)
      GO TO 219
  216 WRITE (6,217)  DSMM(ICUT),PHIA(ICUT)
  217 FORMAT(/29H  THE BEAM CENTER ( RADIUS = ,F6.3,58H ) HAS BEEN ADDED
     X BACK TO THE BEAM WITH A PHASE CHANGE OF ,F7.4/)
  219 CALL FIELDS(DSMM(ICUT),RMV(ICUT),PHIA(ICUT))
      IGATE = 1
      GO TO 3623
C     ***********************************************************
C     PROPAGATE  UNROLLED  ANNULUS
  200 IRSTP = IRSTP+1
      NPYP1=NPY+1
      IF(.NOT. INIT) GO TO 232
      READ(IN,RPROP)
      ABC(1,IRSTP,8) = DELZR
      ABC(2,IRSTP,8) = DELZTH
      IF(ABC(2,IRSTP,8).EQ.0.0) ABC(2,IRSTP,8) = DELZR
      ABC(3,IRSTP,8) = WINDOX
      ABC(4,IRSTP,8) = WINDOK
  232 WRITE (6,234)   (ABC(IST,IRSTP,8),IST=1,4),ANGX,ANGY
  234 FORMAT(///59H     DELZR      DELZTH    WINDOX    WINDOK      ANGX
     X  ANGY, / 6F10.4//)
      CALL RSTEP(ABC(1,IRSTP,8),ABC(2,IRSTP,8),ABC(3,IRSTP,8),ABC(4,
     1IRSTP,8),ANGX,ANGY,Y)
      CALL POWR(CU,X,NPTS,NPYP1)
      IGATE = 1
      GO TO 3623
```

```
C     *************************************************************
C       APPLY AXICON
  190 IAX=IAX+1
      IF(.NOT.INIT) GO TO 191
      READ(IN,AXICON)
      CPH(IAX)=CAPH
      XPND(IAX)=EXPAND
      RCURVE(IAX)=ROC
      TLT(IAX)    = TILT
      DSP(IAX)    = DISP
  191 CALL AXICN(CPH(IAX),XPND(IAX),RCURVE(IAX),DSP(IAX),TLT(IAX),Y)
      GO TO 999
C     *************************************************************
C       APPLY SPIDER OBSCURATION TRANSMISSION FUNCTION TO THE COMPLEX
C       FIELD
  180 READ(IN,SPIDR)
      WRITE(6,181)WIDTH,NSPD,XSPC,YSPC,DIM,(THETA(ISPD),ISPD=1,NSPD)
  181 FORMAT(24H0  SPIDER MODEL APPLIED(,/,15H  STRUT WIDTH =,G12.3,15H
     1NO. OF STRUTS=,I3,12H X-Y CENTER=,G12.4,1H,,G12.4,
     215H HUB DIAMETER =,G12.4/9H THETAS =,6G12.4)
      NSPD = MIN0(NSPD,6)
  182 CALL SPIDER(WIDTH,THETA,NSPD,XSPC,YSPC,DIM)
      IGNAL=4
      GO TO 999
C     *************************************************************
C       WRITE COMPLEX FIELD ON PUNCH CARDS
  160 WRITE(6,163)
  163 FORMAT(36H0 CU HAS BEEN WRITTEN ON PUNCH CARDS)
      WRITE(4,164)(GNOT(ICNTL,I),I=1,20)
  164 FORMAT(20A4)
      DO 161 J=1,NPY
      DO 161 I=1,NPTS,2
      IREF =(J-1)*NPTS
      DUM1=REAL(CU(IREF+I))
      DUME1=AIMAG(CU(IREF+I))
      DUM2=REAL(CU(IREF+I+1))
      DUME2=AIMAG(CU(IREF+I+1))
  161 WRITE(4,162)X(I),X(J),DUM1,DUME1,X(I+1),X(J),DUM2,DUME2
  162 FORMAT(2F8.2,2E12.4,2F8.2,2E12.4)
      IGATE = 1
      GO TO 3623
C     *************************************************************
C       APPLY SINUSOIDAL PHASE VARIATION TO COMPLEX FIELD
  420 IF (.NOT.INIT) GO TO 421
      READ (IN,SINDEN)
  421 WRITE (6,422) NBEAM,AWL
  422 FORMAT (/48H   SINUSOIDAL DENSITY FIELD APPLIED TO THE BEAM /28H
     X     = OF CYCLES PER XCALC ,I5,26H    AMP/WL OF VARIATIONS  =,F7.3 )
      A5 = 2.*3.141592 * AWL
      A6 = 2.   * 3.141592 * NBEAM /(NPTS*(X(2)-X(1)))
      DO 423 I=1,NPTS
      CFACTT=    CEXP(CMPLX(0., A5 * SIN (A6*X(I))))
      DO 423 J=1,NPY
      IJ = I + (J-1)*NPTS
  423 CU(IJ) = CU(IJ)*CFACTT
      GO TO 999
C     *************************************************************
C       APPLY GDL CAVITY TO COMPLEX FIELD
   10 ICAV=ICAV+1
      IF(.NOT. INIT) GO TO 11
      READ(IN,CAVTY1)
      IDIR(1,ICAV) = NCAVNO
      IDIR(2,ICAV) = ICR
      IDIR(3,ICAV) = NSTE
      IDIR(4,ICAV) = NPLT
      ZLI(ICAV)=ZPROPI
      ZLO(ICAV)=ZPROPO
   11 NEWCAV = 0
      NCS = MAX0(IDIR(1,ICAV),NCT)
      IF(NCS.GT.NCT) NEWCAV=1
      NCT = NCS
      WRITE (6,12) IDIR(1,ICAV),IDIR(2,ICAV),IDIR(3,ICAV)
```

```
   12 FORMAT (///16H   CAVITY NUMBER,I3,17H       DIRECTION ,I2 ,29H
     X   PROPAGATING PARAMETER ,I2 /)
      WRITE(6,15) ZLI(ICAV), ZLO(ICAV)
   15 FORMAT(48H0ADDITIONAL PROPAGATION DISTANCES AT CAVITY ENDS/
     X 1X,4HZLI=,G12.5,6X,4HZLO=,G12.5)
      CALL CAVITY(IDIR(1,ICAV),IDIR(2,ICAV),NEWCAV,INIT,IDIR(3,ICAV),IN,
     X  RESTRT, IDIR(4,ICAV),ZLI(ICAV),ZLO(ICAV))
      IF(IDIR(3,ICAV).LE.3) INT=1
      GO TO 999
C     ****************************************************************
C     READ AND/OR WRITE COMPLEX FIELD ON DIRECT ACCESS FILE
  100 NDS = NDS + 1
      IF (.NOT.INIT) GO TO 101
      READ(IN,DISKIT)
      IDSK(1,NDS) = IREAD
      IDSK(2,NDS) = IWRITE
      IDSK(3,NDS) = IORD
      IDSK(4,NDS) = IADD
      GO TO 107
  101 IREAD = IDSK(1,NDS)
      IDSK(2,NDS) = IDSK(2,NDS) + IDSK(4,NDS)
      IWRITE = IDSK(2,NDS)
      IORD = IDSK(3,NDS)
  107 IF (IREAD.EQ.0.OR.IORD.EQ.-1) GO TO 102
      READ (IREAD) (CU(IZ),IZ=1,NOB),X,DRX,DRY,NITER
      WRITE(6,105)IREAD
  105 FORMAT(//10X,26HCU HAS BEEN READ FROM UNIT,I3//)
      REWIND IREAD
  102 IF (IWRITE.EQ.0) GO TO 103
      WRITE (IWRITE) (CU(IZ),IZ=1,NOB),X,DRX,DRY,NITER,SAVE
      WRITE(6,106)IWRITE
  106 FORMAT(//10X,27HCU HAS BEEN WRITTEN ON UNIT,I3//)
      REWIND IWRITE
  103 IF (IREAD.EQ.0.OR.IORD.EQ.1) GO TO 999
      READ (IREAD) (CU(IZ),IZ=1,NOB),X,DRX,DRY,NITER
      WRITE(6,105)IREAD
      REWIND IREAD
      GO TO 999
C     ****************************************************************
C     APPLY AERODYNAMIC WINDOW TO COMPLEX FIELD
  340 WRITE (6,341)
  341 FORMAT (//74H  AERO WINDOW MODEL HAS BEEN APPLIED...RMS PHASE DIST
     XORTION IS THE MODEL  /)
      CALL AEROW(CU,NPTS,NPY)
      GO TO 999
C     ****************************************************************
C     APPLY FIELD SCALING FACTOR
  350 MLT=MLT+1
      IF (.NOT. INIT) GO TO 351
      READ (IN,MULT)
      ABC(1,MLT,9)=TRANS
      ABC(2,MLT,9)=XMAG
  351 WRITE(6,352) ABC(1,MLT,9),ABC(2,MLT,9)
      STRANS = SQRT(ABC(1,MLT,9))/ABC(2,MLT,9)
  352 FORMAT (/43H THE FIELD HAS BEEN SCALED BY THE FACTORS  ,2F8.3/)
      DO 353 I=1,NOB
  353 CU(I) = CU(I)*STRANS
      DO 357 I = 1,NPTS
  357 X(I) = X(I) * ABC(2,MLT,9)
      RMIRR = ABC(1,MLT,9)
      IGNAL = 5
      GO TO 999
C     ****************************************************************
C     MAKE PRINTER PLOTS OF COMPLEX FIELD
   80 IPTT=IPTT+1
      IF (.NOT.INIT) GO TO 82
      READ(IN,PLOT)
      READ (5,1243) TITLE
      DO 83 NW=1,20
   83 APLT(IPTT,NW)=TITLE(NW)
   82 WRITE (6,84) (APLT(IPTT,NW),NW=1,20)
```

```
   84 FORMAT (1H1,30X,20A4 //)                                          GUL      509
      IF(RADPLT.EQ.0.0) CALL IPLOT (1111)                               GUL      510
      IF(RADPLT.NE.0.0) CALL IPLOT (11111)                              GUL      511
      IF(.NOT.INIT)GO TO 98                                             GUL      512
      GO TO 1000                                                        GUL      513
C     ***************************************************************   GUL      514
C     FLIP THE COMPLEX FIELD ABOUT THE Y-AXIS                           GUL      515
  360 NP = NPTS / 2                                                     GUL      516
      WRITE (6,361)                                                     GUL      517
  361 FORMAT(/52H  THE FIELD HAS JUST BEEN FLIPPED ABOUT THE Y-AXIS !/) GUL      518
      DO 362 J=1,NPY                                                    GUL      519
      DO 362 I=1,NP                                                     GUL      520
      I2 = I + (J-1) * NPTS                                             GUL      521
      I3 = 1 - I + NPTS * J                                             GUL      522
      CUD = CU (I2)                                                     GUL      523
      CU(I2) = CU(I3)                                                   GUL      524
  362 CU(I3) = CUD                                                      GUL      525
      GO TO 999                                                         GUL      526
C    ************************************************************      SOU77CY1  169
  365 IF(NPY.NE.NPTS) GO TO 999                                         SOU77CY1  170
      NP=NPTS/2                                                         SOU77CY1  171
      WRITE(6,366)                                                      SOU77CY1  172
  366 FORMAT(/46H   THE FIELD HAS BEEN FLIPPED ABOUT THE X-AXIS/)       SOU77CY1  173
      DO 367 I=1,NPTS                                                   SOU77CY1  174
      DO 367 J=1,NP                                                     SOU77CY1  175
      I2=I+(J-1)*NPTS                                                   SOU77CY1  176
      I3=I+NOB-J*NPTS                                                   SOU77CY1  177
      CUD=CU(I2)                                                        SOU77CY1  178
      CU(I2)=CU(I3)                                                     SOU77CY1  179
  367 CU(I3)=CUD                                                        SOU77CY1  180
      GO TO 999                                                         SOU77CY1  181
C     ***************************************************************   GUL      527
C     APPLY MIRROR TRANSMISSION FUNCTION TO THE COMPLEX FIELD           GUL      528
   20 IMIR = IMIR+1                                                     GUL      529
      IF(.NOT. INIT) GO TO 21                                           GUL      530
      READ(IN,MIROR)                                                    GUL      531
      ABC(1,IMIR,2)= ANGXX                                              GUL      532
      ABC(2,IMIR,2)= ANGYY                                              GUL      533
      ABC(3,IMIR,2)= RADC                                               GUL      534
      ABC(4,IMIR,2)= DIAOUT/2.                                          GUL      535
      ABC(5,IMIR,2)= DIAIN/2.                                           GUL      536
      ABC(6,IMIR,2)= XMPOS                                              GUL      537
      ABC(7,IMIR,2)= YMPOS                                              GUL      538
      ABC(8,IMIR,2)= RMIR                                               GUL      539
      ABC(9,IMIR,2)= DELTA                                              GUL      540
      ABC(10,IMIR,2)= DISTF                                             GUL      541
      ABC(11,IMIR,2)= RANULS                                            GUL      542
      ABC(10,IMIR,4)=DOUTY/2.                                           SUAPR     27
      ABC(11,IMIR,4)=DINY /2.                                           SUAPR     28
      ABC(12,IMIR,2)= PHIAST                                            C10ASTG    6
   21 CALL MIRROR(ABC(1,IMIR,2),ABC(2,IMIR,2),ABC(3,IMIR,2),ABC(4,IMIR,2 GUL      543
     1),ABC(5,IMIR,2),ABC(6,IMIR,2),ABC(7,IMIR,2),ABC(8,IMIR,2),        GUL      544
     2 ABC(9,IMIR,2),ABC(10,IMIR,2),ABC(11,IMIR,2),ABC(10,IMIR,4),      SUAPR     29
     3 ABC(11,IMIR,4),ABC(12,IMIR,2))                                   C10ASTG    7
      RAPTR=ABC(4,IMIR,2)                                               GUL      546
      WRITE(6,23) (ABC(IMR,IMIR,2),IMR=1,3),(ABC(IMR,IMIR,2),IMR=6,11)  SUAPR     31
   23 FORMAT(///8H ANGXX =,G12.4,8H ANGYY =,G12.4,17H RADIUS OF CURV =,G GUL      548
     X12.4/                                                          43H SUAPR     32
     X POSITION OF MIRROR W.R.T. OPTICAL AXIS = (,F6.3,1H,,F6.3,1H) /   GUL      550
     X22H MIRROR REFLECTIVITY =,F8.3,5X/                                GUL      551
     X37H MIRROR SPHERICAL DISTORTION FACTOR =,E12.4/                   GUL      552
     X37H MIRROR FLUX DEP. DISTORTION FACTOR =,E12.4/,                  GUL      553
     X37H OUTSIDE RADIUS OF ANNULAR BEAM      =,E12.4/)                 GUL      554
      IF (ABC(10,IMIR,2).GT.-10.) GO TO 8316                            EDIPWR     1
      WRITE(6,3913)                                                     EDIPWR     2
 3913 FORMAT(58H EDI LOSS ACCOUNTED FOR IN ASSOCIATED MIRROR CALCULATION EDIPWR     3
     XS  )                                                              EDIPWR     4
      GO TO 3623                                                        EDIPWR     5
 8316 CONTINUE                                                          EDIPWR     6
      IGNAL=2                                                           GUL      555
      IF(ABC(4,IMIR,2).LE.0.0.AND.ABC(5,IMIR,2).EQ.0.0)IGNAL=5          GUL      556
      RMIRR=ABC(8,IMIR,2)                                               GUL      557
      GO TO 999                                                         GUL      558
```

```
C    ***********************************************************   GOL       559
C       APPLY TRANSMISSION FUNCTION OF A QUIESCENT THERMAL GRADIENTS     GOL       560
C       NEAR A MIRROR SURFACE                                            GOL       561
  170 ITHRML = ITHRML + 1                                                GOL       562
      IF(.NOT. INIT) GO TO 171                                           GOL       563
      READ(IN,THRML)                                                     GOL       564
      ABC(1,ITHRML,7)= ALPHAM                                            GOL       565
      ABC(2,ITHRML,7)= CONMIR                                            GOL       566
      ABC(3,ITHRML,7)= ALPHAG                                            GOL       567
      ABC(4,ITHRML,7)= RHOGAS                                            GOL       568
      ABC(5,ITHRML,7)= TAU                                               GOL       569
      ABC(6,ITHRML,7)= TIN                                               GOL       570
      ABC(7,ITHRML,7)= REFMIR                                            GOL       571
      ABC(8,ITHRML,7)= CONGAS                                            GOL       572
  171 CALL THERML(ABC(1,ITHRML,7),ABC(2,ITHRML,7),ABC(3,ITHRML,7),ABC(4, GOL       573
     1ITHRML,7),ABC(5,ITHRML,7),ABC(6,ITHRML,7),ABC(7,ITHRML,7),         GOL       574
     2ABC(8,ITHRML,7))                                                   GOL       575
      IGNAL=1                                                            GOL       576
      GO TO 999                                                          GOL       577
C    ***********************************************************   GOL       578
C       APPLY PROPAGATION ALGORITHM TO COMPLEX FIELD                     GOL       579
   30 ISTEP = ISTEP+1                                                    GOL       580
      IF(.NOT. INIT) GO TO 32                                            GOL       581
      READ(IN,PROPGT)                                                    GOL       582
      IF(IIPS.GT.1)IIFG=2                                                GOL       583
      ABC(1,ISTEP,3) = DELZ                                              GOL       584
      ABC(2,ISTEP,3) = ROCURV                                            GOL       585
      ABC(3,ISTEP,3) = WINDOX                                            GOL       586
      ABC(4,ISTEP,3) = WINDOK                                            GOL       587
      ABC(5,ISTEP,3) = IIFG                                              GOL       588
      ABC(6,ISTEP,3) = IITH                                              GOL       589
      ABC(7,ISTEP,3) = IIPS                                              GOL       590
   32 IFG = ABC(5,ISTEP,3)+.001                                          GOL       591
      ITH = ABC(6,ISTEP,3)+.001                                          GOL       592
      IPS = ABC(7,ISTEP,3)+.001                                          GOL       593
      WRITE (6,34)  (ABC(IST,ISTEP,3),IST=1,4),IFG,ITH,IPS,ANGX,ANGY     GOL       594
   34 FORMAT(/// 91H    DELZ     RAD CURV    WINDOX     WINDOK     IFG   GOL       595
     X  ITH        IPS         ANGX           ANGY   / 4F10.4,I6,5X,I6,5X,I6, GOL  596
     X5X,2F10.5//)                                                       GOL       597
      ICORE = 0                                                          GOL       598
      IF (IFG.LT.-5) GO TO 31                                            GOL       599
      IF (ABS(ABC(2,ISTEP,3)).LT..5) ABC(2,ISTEP,3)=RADCUR               GOL       600
  462 CALL STEP(ABC(1,ISTEP,3),ABC(2,ISTEP,3),               ABC(3,ISTEP GOL       601
     1,3),ABC(4,ISTEP,3),IFG,ITH,IPS,ANGX,ANGY,0,ICORE)                  GOL       602
      IF (ICORE.EQ.0) INT = 1                                            GOL       603
      MSTEP=1                                                            GOL       604
      GO TO 999                                                          GOL       605
   31 IF(INT.EQ.0)WRITE(6,319)                                           GOL       606
  319 FORMAT(50H0 ENTERING CORE BEFORE STEP CALLED! CALCULATIONS STOPPED GOL       607
     X)                                                                  GOL       608
      IF(INT.EQ.0)STOP                                                   GOL       609
C     CALL CORE(ABC(1,ISTEP,3),ITH,0)                                    GOL       610
      ICORE=1                                                            GOL       611
      GO TO 462                                                          GOL       612
C    ***********************************************************   GOL       613
C       APPLY APERTURE TRANSMISSION FUNCTION TO COMPLEX FIELD            GOL       614
   40 IAP = IAP+1                                                        GOL       615
      IF(.NOT. INIT) GO TO 41                                            GOL       616
      READ(IN,APTUR)                                                     GOL       617
      ABC(1,IAP,4)= DOUT/2.                                              GOL       618
      ABC(2,IAP,4)= DIN/2.                                               GOL       619
      ABC(3,IAP,4)= XPOS                                                 GOL       620
      ABC(4,IAP,4)= YPOS                                                 GOL       621
      ABC(5,IAP,4)=YOUT/2.                                               SQAPR      33
      ABC(6,IAP,4)=YIN /2.                                               SQAPR      34
   41 IF(DOUT.LT.0.0.AND.DIN.LT.0.0)                                     GOL       622
     1CALL SLIVER(ABC(1,IAP,4),ABC(2,IAP,4),ABC(3,IAP,4),ABC(4,IAP,4))   GOL       623
      IF(DOUT.GE.0.0.AND.DIN.GE.0.0) CALL APRTR(ABC(1,IAP,4),ABC(2,IAP,4 SQAPR      35
     X),ABC(3,IAP,4),ABC(4,IAP,4),ABC(5,IAP,4),ABC(6,IAP,4))             SQAPR      36
      IF(DOUT.GT.0.0.AND.DIN.GE.0.0)RAPTR=ABC(1,IAP,4)                   GOL       626
      IGNAL=4                                                            GOL       630
      GO TO 999                                                          GOL       631
```

```
C     ****************************************************************
C       APPLY THERMAL BLOOMING TRANSMISSION FUNCTION TO COMPLEX FIELD
   50 IDK = IDK+1
      IF ( .NOT. INIT ) GO TO 51
      READ(IN,BLOOM)
      ABC(1,IDK,5) = ALFA
      ABC(2,IDK,5) = SCP
      ABC(3,IDK,5) = T
      ABC(4,IDK,5) = RHO
      ABC(5,IDK,5) = ZLEN
      ABC(6,IDK,5) = NSTEPS
      ABC(7,IDK,5) = INPT
      ABC(8,IDK,5) = NPROP
      ABC(9,IDK,5) = AXIAL
      ABC(10,IDK,5)= DT
   51 NSTEPS = ABC(6,IDK,5)+.0001
      INPT=ABC(7,IDK,5)+.0001
      I22T=ABC(8,IDK,5)+.0001
      CALL TBLOOM(ABC(1,IDK,5),ABC(2,IDK,5),ABC(3,IDK,5),ABC(4,IDK,5),
     XABC(5,IDK,5),NSTEPS,INPT,I22T,ABC(9,IDK,5),ABC(10,IDK,5))
      GO TO 999
C     ****************************************************************
C       INTERPOLATE FEEDBACK FIELD FROM RESONATOR MODE FOR USE IN NEXT
C       ITERATION
   60 IF(.NOT.INIT.AND..NOT.WHY) GO TO 61
      IF(.NOT.INIT)  GO TO 67
      READ(IN,CUTOUT)
      ABC(1,1,1) = DIBEAM
      ABC(2,1,1) = OVRLAP
      ABC(3,1,1) = DXXR
      ABC(4,1,1) = DYYR
      ABC(5,1,1) = AVCUSH
      IGOL(99) = IABS(MAXI1)
   67 DCIBM = ABC(2,1,1)*ABC(1,1,1)/2.
      DIBEAM = ABC(1,1,1)
      XDEL = DCIBM/NPTS*2.
      XK(1) = -DCIBM+XDEL/2.
      DO 62 IGN=2,NPTS
   62 XK(IGN) = XK(IGN-1)+XDEL
      TXY(1) = X(2) - X(1)
      TXY(2) = X(2) - X(1)
      TXY(3) = NPY
      TXY(4) = NPTS
      DO 64 MSP=1,NPY
   64 TXY(4+MSP) = X(MSP)+DRY
      NPY4=NPY+4
      DO 640 MST=1,NPTS
  640 TXY(NPY4 + MST) = X(MST)+DRX
   61 AVC = ABC(5,1,1)
      POWA = 0.
      DX2=(X(2)-X(1))/2.
      DB2=DCIBM
      DO 621 J=1,NPY
      IF (ABS(X(J))-DX2.GT.DB2) GO TO 621
      FCY=1.0
      IF (ABS(X(J))+DX2.LT.DB2) GO TO 627
      FCY=(DB2-(ABS(X(J))-DX2))/DX2/2.
  627 J1 = (J-1) * NPTS
      DO 620 I=1,NPTS
      IF (ABS(X(I))-DX2.GT.DB2) GO TO 620
      FCX=1.0
      IF (ABS(X(I))+DX2.LT.DB2) GO TO 628
      FCX=(DB2-(ABS(X(I))-DX2))/DX2/2.
  628 IX = J1 + I
      POWA = POWA + CU(IX) * CONJG(CU(IX)) * FCX * FCY
  620 CONTINUE
  621 CONTINUE
      POWA = POWA * (X(2)-X(1))**2 / 1000.
      MAAA= 0
      IZ=0
      IF (NPTS.NE.NPY) IZ=1
```

```
      POWB = 0.                                                          APH26      12
      DO 63 MY=1,NPY                                                     GOL       684
      YINP = XK(MY) + ABC(4,1,1)                                         GOL       685
      DO 63 MX=1,NPTS                                                    GOL       686
      MAAA=MAAA+1                                                        GOL       687
      XINP = XK(MX) + ABC(3,1,1)                                         GOL       688
      CALL INTERP(TXY,XINP,YINP,CU,2,CFFL(MAAA),IZ)                      APH26      13
   63 POWB = POWB + CFFL(MAAA)*CONJG(CFFL(MAAA))                         APH26      14
      POWB = POWB * (XK(2)-XK(1))**2 / 1000.                             APH26      15
      FXFLT = SQRT(POWA/POWB)                                            APH26      16
      DO 623 IX = 1,NOB                                                  APH26      17
  623 CFFL(IX) = CFFL(IX)*FXFLT                                          APH26      18
      WRITE (6,624) POWB,POWA                                            APH26      19
  624 FORMAT (/10X,28HFIELD ADJUSTED FROM POWER OF ,F8.2,4H TO ,F8.2/)   APH26      20
      IF (CUSMF.NE.0.) GO TO 5924                                        CYCLE9      6
      IZ=0                                                               CYCLE9      7
      DO 5843 IX=1,NPTS                                                  CYCLE9      8
      X(IX) = XK(IX)                                                     CYCLE9      9
      DO 5843 IXY=1,NPY                                                  CYCLE9     10
      IZ = IZ + 1                                                        CYCLE9     11
 5843 CU(IZ) = CFFL(IZ)                                                  CYCLE9     12
      GO TO 999                                                          CYCLE9     13
 5924 CONTINUE                                                           CYCLE9     14
      IF(ICAV.GT.0) GO TO 691                                            GOL       690
      FMAX=0.                                                            GOL       691
      DO 692 IM=1,NOB                                                    GOL       692
      FMAG=CABS(CFFL(IM))                                                GOL       693
      IF(FMAG.LT.FMAX) GO TO 692                                         GOL       694
      FMAX=FMAG                                                          GOL       695
      INDX=IM                                                            GOL       696
  692 CONTINUE                                                           GOL       697
      DO 693 IM=1,NOB                                                    GOL       698
  693 CFFL(IM)=CFFL(IM)/FMAX                                             GOL       699
      WRITE(6,6641) FMAX                                                 GOL       700
 6641 FORMAT(//47H  CUTOUT FIELD AMPLITUDES HAVE BEEN DIVIDED BY ,       GOL       701
     X  F8.4,//)                                                         GOL       702
  691 CONTINUE                                                           GOL       703
      WRITE (7) (CU(IZ),IZ=1,NOB)                                        GOL       704
      REWIND 7                                                           GOL       705
      IF(.NOT.RESTRT.AND.INIT)GO TO 630                                  GOL       706
      READ (8) (CFIL2(IZ),IZ=1,NOB),XDUM,DUUM2,DUUM3,NDUUM,SAVE          GOL       707
      REWIND 8                                                           GOL       708
  630 SUMERR=0.0                                                         GOL       709
      ICNT=0                                                             GOL       710
      NWTA=NPTS/16                                                       GOL       711
      NWTB=NPTS/4                                                        GOL       712
      NWTC=NPTS-NWTB                                                     GOL       713
      NWTD=NPTS/2                                                        GOL       714
      WRITE(6,653)                                                       GOL       715
  653 FORMAT(44H0CUTOUT FIELD COMPARISON TO DETERMINE AVGAIN/)           GOL       716
      WRITE(6,71)                                                        GOL       717
   71 FORMAT(10H0 POINT   ,6X,12H  CURRENT    ,4X,12H  PREVIOUS  ,4X,12H GOL       718
     X  PERCENT  /10H  TESTED  ,6X,12H   VALUE     ,4X,12H   VALUE    ,  GOL       719
     X4X,9H   CHANGE//)                                                  GOL       720
      ICEKS=0                                                            GOL       721
      DO 65 IABC=NWTB,NWTC,NWTA                                          GOL       722
      ICNT=ICNT+1                                                        GOL       723
      ERRSM=0.                                                           GOL       724
      DUM=CABS(CFFL(IABC+(NWTD-1)*NPTS))                                 GOL       725
      DUME=CABS(CFIL2(IABC+(NWTD-1)*NPTS))                               GOL       726
      IF(.NOT.RESTRT.AND.INIT)DUME=1.0                                   GOL       727
       IF(DUME.NE.0.) ERRSM=(DUM-DUME)/DUME                              GOL       728
       IF(ABS(ERRSM).GT.0.10)ICEKS=1                                     GOL       729
       SUMERR=ERRSM**2+SUMERR                                            GOL       730
       WRITE(6,650)IABC,NWTD,DUM,DUME,ERRSM                              GOL       731
  650 FORMAT(6H CUSM(,I3,1H,,I2,1H),4X,G12.5,4X,G12.5,7X,2PF6.2)         GOL       732
   65 CONTINUE                                                           GOL       733
      IF (ABC(5,1,1).EQ. 0. .OR.(NITER.EQ.0.AND.KAUTO.EQ.0)) GO TO 69    GOL       734
      IF(ABC(5,1,1).GE.0.) GO TO 68                                      GOL       735
      ERRSS=SQRT(SUMERR/ICNT)                                            GOL       736
      AVC = .8 - ERRSS                                                   SOU77CY1  182
       IF(ERRSS.GT.0.6) AVC=0.2                                          GOL       738
       IF (ERRSS.LT..1) AVC = .7                                         SOU77CY1  183
```

```
      WRITE(6,610)ERRSS,AVC
  610 FORMAT(//17X,29HFIELD AVERAGING HAS BEEN USED/10X,10HRMS ERROR=,
     1 F8.4,5X,12HAVCUSM USED=,F8.4//)
   68 CONTINUE
      DO 75 MX=1,NOB
      XAN = CABS(CFFL(MX))
      XAOLD = CABS(CFIL2(MX))
   75 CFFL(MX) = CFFL(MX) * (AVC*XAOLD+(1.-AVC)*XAN) / XAN
   69 MY = NITER+1
      READ(7) (CU(IZ),IZ=1,NOB)
      REWIND 7
      WRITE(6,663)
  663 FORMAT(12X,33HCONVERGENCE TEST FIELD COMPARISON/)
      ICEK=0
      SMERR=0.0
      ICNT=0
      WRITE(6,71)
      DO 660 IABC=NWTB,NWTC,NWTA
      ICNT=ICNT+1
      ERR=0.
      DUM=CABS(CU(IABC+(NWTD-1)*NPTS))
      DUME=SAVE(ICNT)
      SAVE(ICNT)=DUM
      IF(.NOT.RESTRT.AND.INIT)DUME=1.0
      IF(DUME.NE.0.) ERR=(DUM-DUME)/DUME
      IF(ABS(ERR).GT.0.02)ICEK=1
      SMERR=ERR**2+SMERR
      WRITE(6,661)IABC,NWTD,DUM,DUME,ERR
  661 FORMAT(4H CU(,I3,1H,,I2,1H),4X,G12.5,4X,G12.5,7X,2PF6.2)
  660 CONTINUE
      IF(ICEKS.EQ.1)ICEK=1
      ERRSS=SQRT(SMERR/ICNT)
      WRITE(6,662)ERRSS
  662 FORMAT(/15X,18HRMS ERROR FOR CU =,F8.4/)
      WRITE(8) (CFFL(IZ),IZ=1,NOB),XK,ABC(3,1,1),ABC(4,1,1),MY,SAVE
      REWIND 8
      WRITE (6,66) (ABC(JVCX,1,1),JVCX=1,5)
   66 FORMAT ( //82H     INTERPOLATIONS FOR THE FIELD OVER DIBEAM*OVRLAP
     X   HAVE JUST BEEN PERFORMED     /59H     BEAM DIA      OVERLAP     XPO
     XS      YPOS     FIELD AVERAGE    / 2X,5G12.5 // )
      WHY = .FALSE.
      GO TO 999
C     ****************************************************************
C     INCREASE THE NUMBER OF GRID POINTS FOR COMPLEX FIELD
  150 READ(IN,REGRID)
      NPTSS = NPTS
      NPYS = NPY
      IF(NGRD.GT.NPTS)GO TO 151
      GO TO 734
  151 CALL RGRD(NGRD)
      NOB = NPTS*NPY
      WRITE(6,152)NPTSS,NPYS,NPTS,NPY
  152 FORMAT(///5X,21H YOUR ORIGINAL FIELD(,I4,1H*,I3,39H) HAS BEEN REGR
     1IDDED TO A LARGER SIZE (,I4,1H*,I3,45H) TO GIVE THE FIELD MORE ROO
     2M TO DO ITS THING///)
      GO TO 999
C     ****************************************************************
C     RESONATOR CONVERGENCE TEST
   70 NITER = NITER+1
      WRITE (6,605) NITER
  605 FORMAT(////39H   THIS IS THE COMPLETION OF ITERATION ,I3 /)
      IF(INIT.AND..NOT.RESTRT) GO TO 710
      IF(.NOT.INIT) GO TO 720
      GO TO 730
  710 PCVNG=0.0
      GO TO 720
  730 READ (9)(CFIL(IZ),IZ=1,NOB)
      REWIND 9
      PCVNG=0.0
      DO 740 IZ=1,NOB
      PCVNG=PCVNG+CFIL(IZ)*CONJG(CFIL(IZ))
```

```
 740 CONTINUE
     PCVNG=PCVNG*(X(2)-X(1))**2*(NPTS/NPY)
     IF(NREG.EQ.1.OR.NREG.EQ.2)PCVNG=PCVNG/WNUM**2
 720 FERR=1.00
     IF(PCVNG.GT.0.0)FERR=PPW/PCVNG-1.0
     IF (ABS(FERR).GT..007) ICER=1
     PCVNGK=PCVNG/1000.
     WRITE(6,750)PPWK,PCVNGK,FERR
 750 FORMAT(30X,21HFLUX CONVERGENCE TEST//10X,10HNEW FLUX =,0PG11.4,
    X12H  OLD FLUX =,G11.4,9H  ERROR =,F8.4///)
     PCVNG=PPW
     WRITE (9) (CU(IZ),IZ=1,NOB),X,DRX,DRY,NITER
     REWIND 9
     IF (ICER.EQ.0) GO TO 565
     IF (ICAV.GT.0) CALL REGAIN(NCT, NITER)
     IF (NITER-NIT.GE.IGOL(99)) GO TO 1001
     READ (8) (CU(IZ),IZ=1,NOB),X,DRX,DRY,NITER
     REWIND 8
     IF (ICAV.GT.0) GO TO 99
C    RENORMALIZATION OF INPUT FIELD FOR BARE RESONATOR
     IF (.NOT.INIT) GO TO 86
     FMAX = 0.
     DO 87 IX=1,NOB
     IF (CABS(CU(IX)).LE.FMAX) GO TO 87
     FMAX = CABS(CU(IX))
     NP01 = IX
  87 CONTINUE
  86 TEST=CABS(CU(NP01))
     DO 77 IX=1,NOB
  77 CU(IX)    =CU(IX)    /TEST
     GO TO 99
1001 READ (9) (CU(IZ),IZ=1,NOB),X,DRX,DRY
     REWIND 9
     GO TO 1000
C    ****************************************************************
C    CALCULATE UCALC FLUX AND MIRROR AND APERTURE LOSSES
 999 PPW = 0.
     NOB=NPTS*NPY
     DO 78 IZ=1,NOB
  78 PPW=PPW+CU(IZ)*CONJG(CU(IZ))
     PPW=PPW*(X(2)-X(1))**2*(NPTS/NPY)
     IF(NREG.EQ.1.OR.NREG.EQ.2)PPW=PPW/WNUM**2
     PBMIR=PPW
     GO TO (998,997,998,996,997),IGNAL
 997 PBMIR=PPW/RMIRR
     PMIRL=(PBMIR-PPW)/1000.
     PMIRLP=(PBMIR-PPW)/PBMIR*100.
     WRITE(6,995)PMIRL,PMIRLP
 995 FORMAT(17H  MIRROR LOSS   =,G12.4,1H=,F8.2,8H PERCENT)
     IF(IGNAL.EQ.5)GO TO 998
 996 APLOS=(SPPW-PBMIR)/1000.
     APLOSP=(SPPW-PBMIR)/SPPW*100.
     IF(ICNTL.EQ.1)GO TO 998
     WRITE(6,994)APLOS,APLOSP
 994 FORMAT(17H  APERTURE LOSS =,G12.4,1H=,F8.2,8H PERCENT)
 998 PPWK=PPW/1000.
     IGNAL=1
     SPPW=PPW
     UCALCP=X(NPTS)-2.*X(1)+X(2)
     IF(MSTEP.NE.1) WRITE(6,79)PPWK,UCALCP
  79 FORMAT(///38H ELEMENT TRANSMISSION FUNCTION APPLIED/8X,12HUCALC FL
    XUX =, G12.4/8X,12HUCALC       =,F8.2 )
     IF(MSTEP.EQ.1)WRITE(6,779)PPWK
 779 FORMAT(///34H PROPAGATION STEP HAS BEEN APPLIED/    8X,12HUCALC FL
    XUX =, G12.4)
     MSTEP=0
3623 IF (IPLOTS.EQ.0) GO TO 3624
     WRITE (6,3645) (GNOT(ICNTL,I),I=1,20)
     CALL IPLOT(IPLOTS)
     IF (IGATE.NE.0) GO TO 3625
3645 FORMAT (25H1 PLOTS AFTER STEP ****** ,20A4, 6H******)
3624 IF(PPW.LE.0.)GO TO 732
```

```
3625 IF(.NOT.INIT)GO TO 98                                            GOL
     GO TO 1000                                                       GOL
 565 WRITE(6,600)NITER                                                GOL
 600 FORMAT(// 120(1H$)//45H                ITERATION IS CONVERGED AFTE GOL
    X R,I4,14H ITERATIONS   //120(1H*)//)                              GOL
     IF(KAUTO.EQ.1)GO TO 98                                           GOL
     GO TO 1000                                                       GOL
 900 RETURN                                                           GOL
 732 WRITE(6,733)                                                     GOL
 733 FORMAT(///81H ALL RIGHT THERES AIN T  NO POWER IN THIS HERE BEAM A GOL
    AND THE REASON WE RE ALL HERE/58H    IS POWER SO THIS JOB IS GOING GOL
    XTO LED AND KILLED QUICK /23H   ***CHECK INPUT***   //)           GOL
     STOP                                                             GOL
 734 WRITE(6,735)NGRD,NPTS                                            GOL
 735 FORMAT(///5X,26H*X*X*X*   VALUES OF NGRD (,I4,12H) AND NPTS (,I4, GOL
    150H) MAKE THIS OPERATION UNNECESSARY OR WRONG *X*X*X*///)        GOL
      STOP                                                            GOL
      END                                                             GOL
```

14. SUBROUTINE INTERP

a. Purpose -- Subroutine INTERP performs linear interpolation on two-dimensional real functions and on the real and imaginary parts of two-dimensional complex functions. Figure 32 describes the subroutine INTERP organization.

b. Relevant formalism -- Consider first the one-dimensional case in Figure 33. Assume the function value f is desired at a point x*, between points x_1 and x_2, with associated function values f_1 and f_2, respectively:

Linear interpolation between f_1 and f_2 yields f as

$$f(x^*) \approx f_1 + \frac{(x - x_1)}{(x_2 - x_1)} (f_2 - f_1) \qquad (108)$$

where the $\approx$ is used since we are approximating f over the subinternal (x_1, x_2).

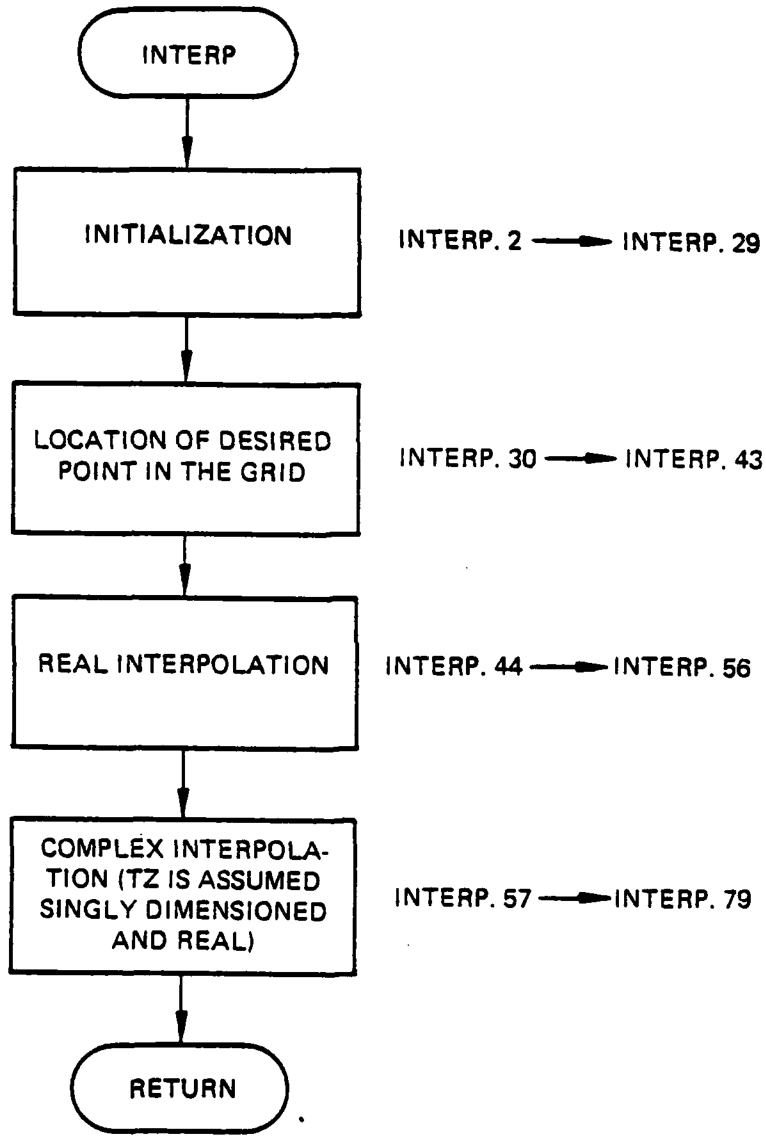


Figure 32. Subroutine INTERP organization.

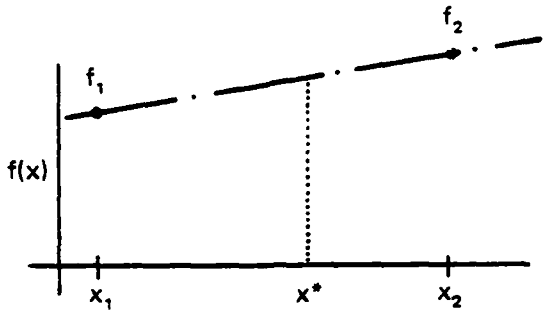
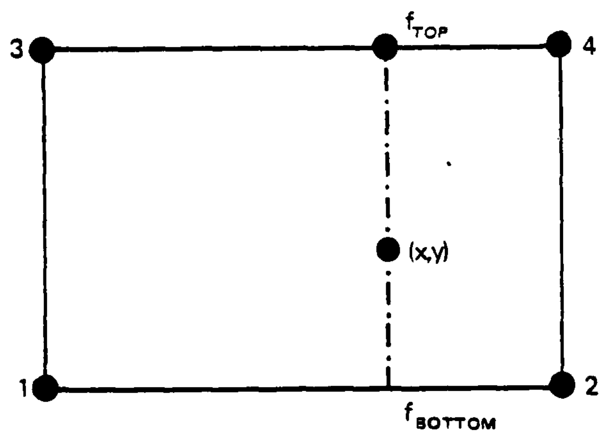


Figure 33. One-dimensional function case.

For the two dimensional case in Figure 34, subroutine INTERP establishes the location of the far corners of the rectangle bounding the desired point (x,y), then linearly interpolates across top and bottom to find the two values at x. It then interpolates between these two points to find the value at (x, y):



$$1 \Leftrightarrow (x_1, y_1) \Leftrightarrow f_1 = f(x_1, y_1) \tag{109}$$

$$2 \Leftrightarrow (x_2, y_1) \Leftrightarrow f_2 = f(x_2, y_1)$$

etc

(x,y) Position of Desired Interpolate

Figure 34. Two-dimensional function case.

$$f(x,y_2) \approx f_{TOP} = f_3 + \frac{(x - x_1)}{(x_2 - x_1)} (f_4 - f_3) \tag{110}$$

$$f(x,y_1) \approx f_{BOTTOM} = f_1 + \frac{(x - x_1)}{(x_2 - x_1)} (f_2 - f_1)$$

$$f(x,y) = f_{BOTTOM} + \left[\frac{(y - y_1)}{(y_2 - y_1)}\right] * (f_{TOP} - f_{BOTTOM})$$

c. Fortran

Arguments:

TXY	= an array containing coordinate information
(XIN, YIN)	= the point at which the function value is desired
TZ	= the function to be interpolated
TYPE	= 1 real = 2 complex
ZZ	= two element array containing the interpolated value. Note: If TZ is real, ZZ must still be dimensioned to 2 in the calling program, then the first element used as the answer.
NSYM	= 1 symmetric, = 0 nonsymmetric Note: Interpolation outside the region of definition of the distribution returns (0.0, 0.0) as the value of the interpolate.

There are no commons and no other subroutines are called.

Computer printouts of subroutine INTERP follow.

SUBROUTINE INTERP 76/176 OPT=1 FIN 4.6+452 04/27/79 12.23.47

```
      SUBROUTINE INTERP(TXY,XIN,YIN,TZ,TYPE,ZZ,NSYM)
C
C  THIS ROUTINE DOES A LINEAR INTERPOLATION ON THE
C  ARRAY TZ TO FIND THE VALUE ZZ AT XIN, YIN
C  THE(X,Y) GRID OF TZ IS CONTAINED IN THE ARRAY TXY
C  AS FOLLOWS:
C      TXY(1) = DX=SPACING BETWEEN X POINTS
C      TXY(2) = DY=SPACING BETWEEN Y POINTS
C      TXY(3) = NY, NO. OF POINTS ALONG Y-AXIS
C      TXY(4) = NX, NO. OF POINTS ALONG X-AXIS
C      TXY(5) = Y(1), MIN. Y VALUE
C      TXY(4+NY) = Y(NY), MAX. Y VALUE
C      TXY(5+NY) = X(1), MIN. X VALUE
C      TXY(4+NY+NX) = X(NX), MAX. X VALUE
```

```
C   ND IS MAX. DIMENSION OF FIRST VARIABLE IN TZ(I,J)
C   TYPE = 1 TZ IS REAL ARRAY
C        = 2 TZ IS COMPLEX ARRAY
      LEVEL 2, TZ,ZZ
      DIMENSION              ZZ(2),TXY(1),TZ(1)
      INTEGER TYPE, COMPLX
      COMPLEX CZZ,CZ1,CZ2,CZ3,CZ4,CZA,CZB
      DATA COMPLX / 2 /
      DX = TXY(1)
      DY = TXY(2)
      NY = TXY(3)+.00001
      NX = TXY(4)+.00001
      ZZ(1) = 0.
      ZZ(2) = 0.
C   TEST TO SEE IF XIN,YIN LIE WITHIN DEFINED TZ REGION
      IF(XIN.LT.TXY(5+NY))  GO TO 1000
      IF(XIN.GT.TXY( 4+NX+NY))  GO TO 1000
      IF(YIN.LT.TXY(5)) GO TO 1000
      IF(YIN.GT.0..AND.NSYM.EQ.1) GO TO 1000
      IF (YIN.GT.TXY(NY+4).AND.NSYM.EQ.0) GO TO 1000
C   FIND POSITION OF (XIN,YIN) IN GRID
      I1 = 1+(XIN-TXY(5+NY))/DX
      J1 = 1+(YIN-TXY(5))/DY
      IF(I1.EQ.NX) I1=I1-1
      IF(J1.EQ.NY.AND.NSYM.EQ.0) J1=J1-1
      SX = (XIN-TXY(I1+4+NY))/DX
      SY = (YIN-TXY(J1+4))/DY
C   FIND TZ VALUES AT I1,I1+1,J1,J1+1
      IF(TYPE.EQ.COMPLX) GO TO 200
C   TZ IS TREATED AS REAL ARRAY
      IJ = I1+NX*(J1-1)
      Z1 =  TZ(IJ)
      Z2 =  TZ(IJ+1)
      IJ = I1+NX*(J1)
      IF (J1.EQ.NY) IJ=IJ-NX
      Z3 =  TZ(IJ)
      Z4 =  TZ(IJ+1)
      ZA = Z1+SX*(Z2-Z1)
      ZB = Z3+SX*(Z4-Z3)
      ZZ(1) = ZA+SY*(ZB-ZA)
      GO TO 1000
  200 CONTINUE
C   TZ IS TREATED AS COMPLEX ARRAY
      IJ = TYPE*(I1+NX*(J1-1)) - 1
      Z1A = TZ(IJ)
      Z1B = TZ(IJ+1)
      CZ1 = CMPLX(Z1A,Z1B)
      Z2A = TZ(IJ+2)
      Z2B = TZ(IJ+3)
      CZ2 = CMPLX(Z2A,Z2B)
      IJ = TYPE*(I1+NX*J1) - 1
      IF (J1.EQ.NY) IJ=IJ-NX*TYPE
      Z3A = TZ(IJ)
      Z3B = TZ(IJ+1)
      CZ3 = CMPLX(Z3A,Z3B)
      Z4A = TZ(IJ+2)
      Z4B = TZ(IJ+3)
      CZ4 = CMPLX(Z4A,Z4B)
      CZA = CZ1+SX*(CZ2-CZ1)
      CZB = CZ3+SX*(CZ4-CZ3)
      CZZ = CZA+SY*(CZB-CZA)
      ZZ(1) = REAL(CZZ)
      ZZ(2) = AIMAG(CZZ)
 1000 RETURN
      END
```

15. SUBROUTINE IPLOT

a. Purpose -- Subroutine IPLOT has two major purposes: One is to create a printer iso-intensity plot. The other is to find the maximum intensity and to print the first title used by subroutine OUTPUT. It also contains the necessary information used by both subroutines OUTPUT and OUTPUR to determine whether a particular slice plot should be printed. Figure 35 describes the subroutine IPLOT organization.

b. Relevant formalism -- The output of this subroutine is an array of one-digit adjacent members with at least one asterisk, which indicates the maximum intensity points. The numbers indicate relative intensities.

c. Fortran

Argument List

The only argument of subroutine IPLOT is the parameter IPLTS which contains the information needed by OUTPUT (and OUTPUR) as well as IPLOT. IPLTS is filled with zero to five digits, each of which is 0 or 1. If it is 0, the indicated plot is not done; if 1, it is plotted. Assuming that the five digits of IPLTS are written ABCDE, the associated plots are:

A: Radial (calls OUTPUR - not available)

B: Iso-intensity

C: X-axis slice plot

D: Diagonal slice plot

E: y-axis slice plot

Common Parameters:

The only common modified is CFIL due to its equivalence with US, the intensity array. The other parameters have then usual meaning including PLOTSG.

Recall: PLOTSG > 0 → intensity slice plots
= 0 → no plots
< 0 → amplitude slice plots

Subroutines called: OUTPUT, OUTPUR

Computer printout of subroutine IPLOT follows.

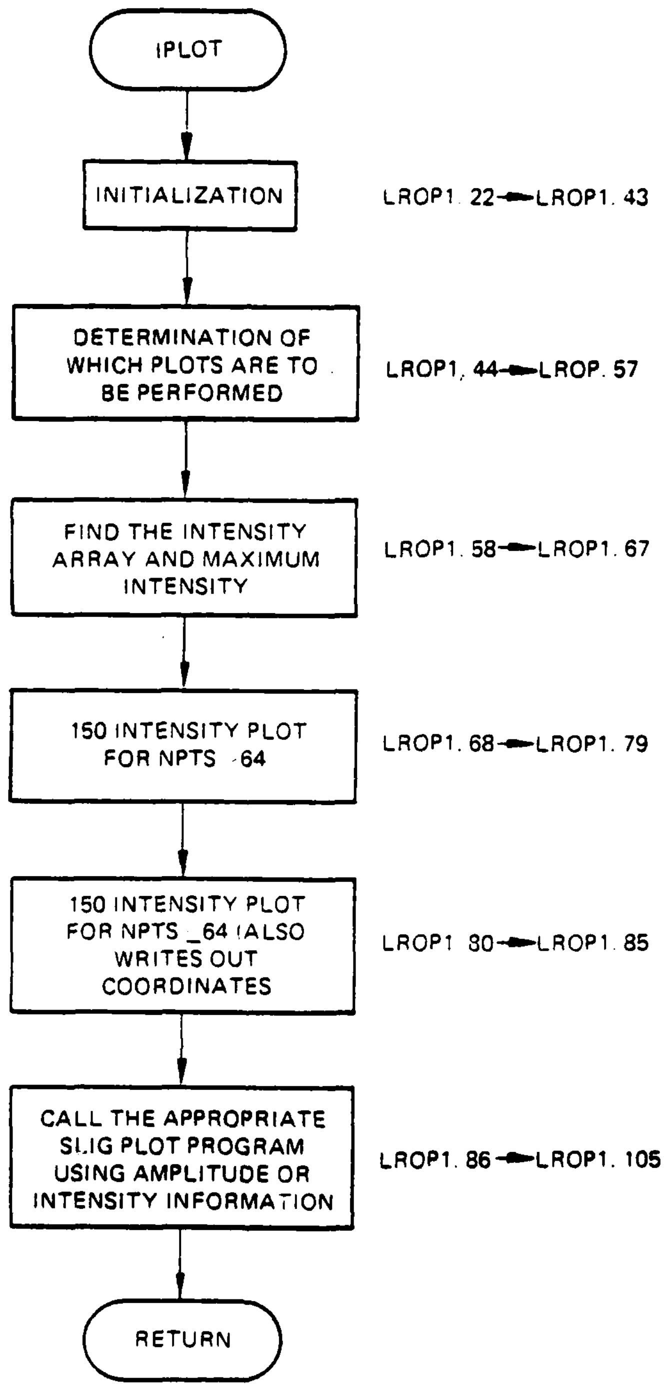


Figure 35. Subroutine IPLOT organization.

```
      SUBROUTINE IPLOT(IPLTS)                                          LHOP1
C     ISO-INTENSITY PRINTER PLOT                                       LHOP1
C     THIS ROUTINE MAKES A PRINTER PLOT OF INTENSITY WHERE THE DIGIT   LHOP1
C     PRINTED * 10 IS DECILE OF PEAK INTENSITY FOR THAT ELEMENT.       LHOP1
      LEVEL 2, CUR,US                                                  LHOP1
      COMMON/MELT/CUR(32768),CFIL(16512),X(128),WL,NPTS,NPY,DRX,DRY    LHOP1
      COMMON/WAY/WNOW,NREG,RAPTH                                       LHOP1
      COMMON /PLTSIG/ PLOTSG                                           LHOP1
      DIMENSION US(16384 ),II(150)                                     LHOP1
      INTEGER   II,BLANK,DOT                                           LHOP1
      LOGICAL ISOIP,XAXIS,DIAG,YAXIS,RADPLT                            LHOP1
      COMPLEX CFIL                                                     LHOP1
      EQUIVALENCE(CFIL( 1 ),US( 1 ))                                   LHOP1
      DATA IBCDR / 4HR    /,IBCDX/4HX    /                             LHOP1
      IF (PLOTSG.EQ.0.) RETURN                                         LHOP1
      IHORX=IBCDX                                                      LHOP1
      RADPLT=.FALSE.                                                   LHOP1
      ISOIP=.FALSE.                                                    LHOP1
      XAXIS=.FALSE.                                                    LHOP1
      DIAG=.FALSE.                                                     LHOP1
      YAXIS=.FALSE.                                                    LHOP1
      IPL=IPLTS                                                        LHOP1
      IF (IPL.LT.10000) GO TO 290                                      LHOP1
      RADPLT = .TRUE.                                                  LHOP1
      IPL = IPL - 10000                                                LHOP1
      IHORX = IBCDR                                                    LHOP1
  290 IF (IPL.LT.1000) GO TO 300                                       LHOP1
      ISOIP=.TRUE.                                                     LHOP1
      IPL = IPL - 1000                                                 LHOP1
  300 IF (IPL.LT.100) GO TO 400                                        LHOP1
      XAXIS = .TRUE.                                                   LHOP1
      IPL = IPL - 100                                                  LHOP1
  400 IF (IPL.LT.10) GO TO 500                                         LHOP1
      DIAG= .TRUE.                                                     LHOP1
      IPL = IPL - 10                                                   LHOP1
  500 IF (IPL.NE.0) YAXIS=.TRUE.                                       LHOP1
      PI=3.141592                                                      LHOP1
      DX=X(2)-X(1)                                                     LHOP1
      XDIM=DX*NPTS                                                     LHOP1
      NOB=NPTS*NPY                                                     LHOP1
      XFACT=1.                                                         LHOP1
      IF (NREG.EQ.1.OR.NREG.EQ.2) XFACT=1./WNOW**2                     LHOP1
      UMAX=0.                                                          LHOP1
      DO 1 J=1,NOB                                                     LHOP1
      US(J)= (CUR(2*J-1)**2 + CUR(2*J)**2) * XFACT                     LHOP1
    1 UMAX=AMAX1(UMAX,US( J ))                                         LHOP1
      IF (.NOT.ISOIP) GO TO 98                                         LHOP1
      UMAXK=UMAX/1000.                                                 LHOP1
      IF(NPY.LE.64)WRITE(6,5) IHORX                                    LHOP1
    5 FORMAT(9X,A1 )                                                   LHOP1
      IF(NPY.LE.64)GO TO 99                                            LHOP1
      DO 4 J=1,NPTS                                                    LHOP1
      DO 2 I=1,NPY                                                     LHOP1
      I2 = J + (I-1)*NPTS                                              LHOP1
    2 II(I)=10.*US( I2)/UMAX                                           LHOP1
    4 WRITE(6,3) (II(I),I=1,NPY)                                       LHOP1
    3 FORMAT(  1X,128I1)                                               LHOP1
      GO TO 98                                                         LHOP1
   99 DO 14 J=1,NPTS                                                   LHOP1
      DO 12 I=1,NPY                                                    LHOP1
      I2 = J + (I-1)*NPTS                                              LHOP1
   12 II(I)=10.*US( I2)/UMAX                                           LHOP1
   14 WRITE(6,13) X(J),(II(I),I=1,NPY)                                 LHOP1
   13 FORMAT(  1X,F10.2,2X,64I1)                                       LHOP1
   98 WRITE(6,6)  XDIM, UMAXK,DRX,DRY                                  LHOP1
    6 FORMAT(10H0 UCALC = ,G11.5,4X,7HIMAX = ,G11.4//24X,              LHOP1
     X    39HTHE CENTER OF THE BEAM IS LOCATED AT  (,F6.3,1H,,F6.3,1H)) LHOP1
```

```
      IF (PLOTSG.GT.0.) GO TO 1500
      IF (.NOT.RADPLT) WRITE (6,7)
    7 FORMAT (
     X90H1AMPLITUDE, PHASE  PLOTTED IN THE X-DIRECTION THROUGH THE CENTE
     XR OF DCALC (J=NPTS/2)
      UMAXA=SQRT(UMAX)
      GO TO 1550
 1500 WRITE (6,786)
  786 FORMAT (
     X90H1INTENSITY, PHASE  PLOTTED IN THE X-DIRECTION THROUGH THE CENTE
     XR OF DCALC (J=NPTS/2)
      UMAXA = UMAX
 1550 IF(NREG.NE.0.AND.PLOTSG.LT.0.) UMAXA=UMAXA*WNOW
      IF(NREG.NE.0.AND.PLOTSG.GT.0.) UMAXA=UMAXA*WNOW**2
      IF (.NOT.RADPLT) CALL OUTPUT(CUR,NPY,NPTS,X,3,UMAXA,XAXIS,DIAG,
     X YAXIS )
      IF (RADPLT) CALL OUTPUR(CUR,NPY,NPTS,X,UMAXA,XAXIS,DIAG,YAXIS)
      RETURN
      END
```

16. SUBROUTINE KINET

a. Purpose -- This subroutine calculates the kinetics and loaded gain in the gas dynamic laser cavity. It is called by GAINXY for either small signal gain calculation (along a single stream tube in the x-direction) or full field loaded gain along several stream tubes. Figure 36 describes the subroutine KINET flow chart.

An intensity field XIC and previous gain field GAN are brought in from GAINXY and are updated by recomputing the kinetics and gain in the cavity as a function of these updated fields. The population rate equations (i.e., the equations showing the rate at which the energy of each vibrational level is changing) are numerically integrated along the x(flow)-direction. This is continued along the x-direction until the end of the calculation region (IXMAX) and is then redone for each stream tube in the y-direction (if full loaded gain is requested by IFIELD $\neq$ 1). The full gain field GAN (I) is then updated.

The assumption is made that the flow area of the cavity is constant through the region of interest for all kinetics calculations.

b. Relevant formalism -- Gain is calculated in the x-direction from nozzle exit plane to the end of the region of interest IXMAX at a constant y value, as shown in Figure 37. This is done along only one mid-cavity stream tube for small signal gain calculation and at every y-value (IY) for the full field loaded gain calculations.

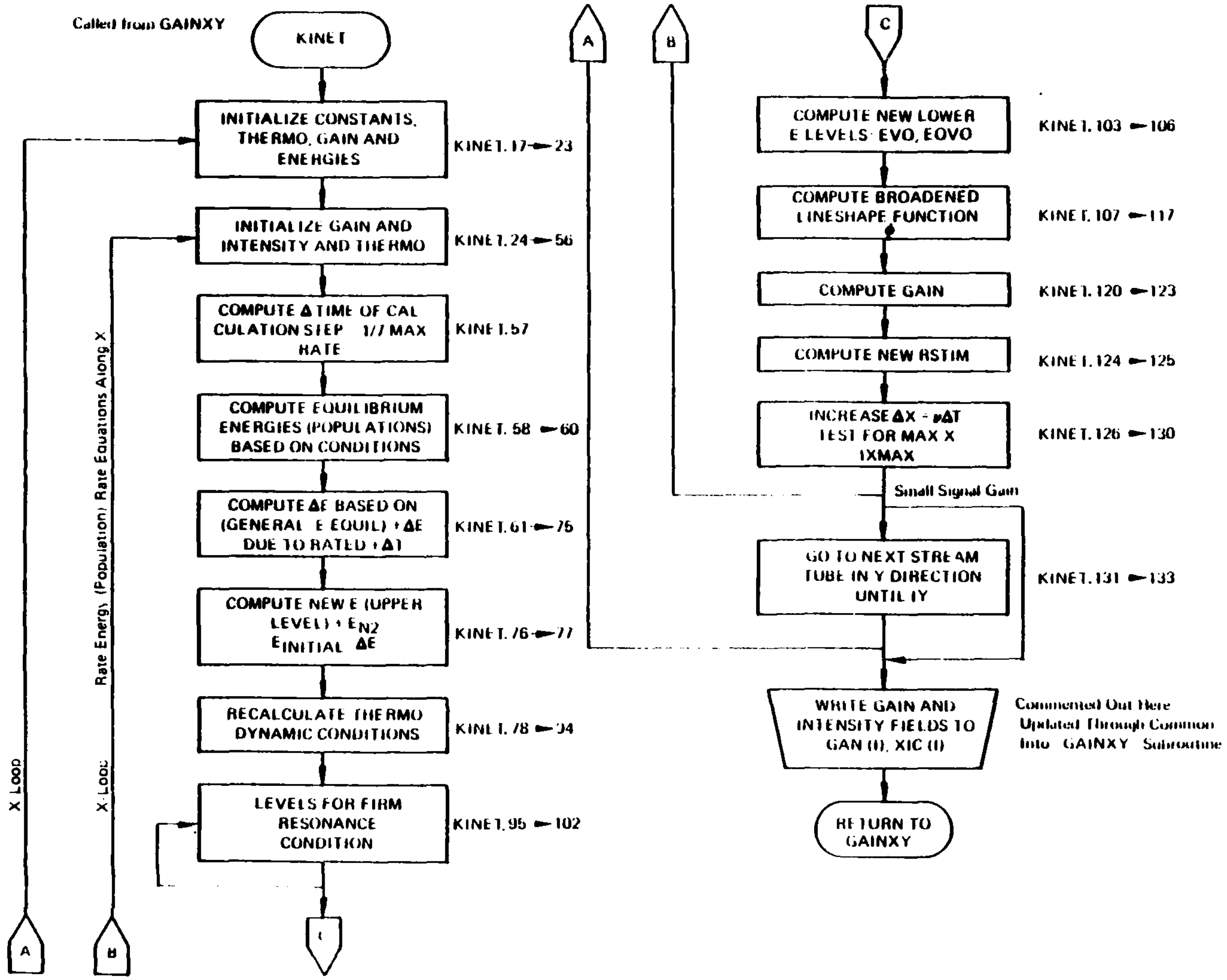


Figure 36. Subroutine KINET flow chart.

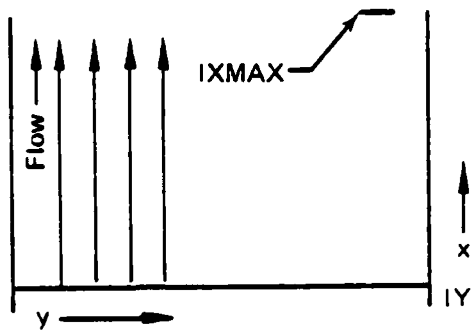


Figure 37. Region of interest IXMAX.

Rate equations are set up for each level which describe the energy in that level.

$$\frac{dE}{dt}\text{ upper} = \left[\frac{dE}{dt}\right]_{V-T} + \left[\frac{dE}{dt}\right]_{V-V} + \left[\frac{dE}{dt}\right]_{S.E.} \tag{111}$$

$$\frac{dE}{dt}\text{ lower} = \left[\frac{dE}{dt}\right]_{V-T} + \left[\frac{dE}{dt}\right]_{V-V} + \left[\frac{dE}{dt}\right]_{S.E.} \tag{112}$$

$$\frac{dE}{dt} N_2 = \left[\frac{dE}{dt}\right]_{V-T} + \left[\frac{dE}{dt}\right]_{V-V} . \tag{113}$$

The energies of each level EN2, EOOV, EVOO and EOVO are updated at each ΔX step, i.e., the ΔE change is computed and the corresponding heat addition (local temperature change) is used to compute the energy in the subsequent step.

The stimulated emission energy rate can be used to determine local intensity change and, hence, gain. Energies of levels are described by population densities n_u and n_L:

$$\left.\frac{dI_\nu}{dS}\right|_{u\neq L} = h\nu \left\{ n_u \bar{N}_{UL}(\nu)_{UL} A_{UL}(\nu) - \left[n_1 \Phi_{LU}(\nu) B_{LU} - n_u N_{UL}(\nu) B_{LU} \right] I\nu \right\} \tag{114}$$

where I_ν is the specific intensity at the frequency ν: n_U is the population density of the upper level; n_L, that of the lower level; A_{UL} the Einstein coefficient for spontaneous emission; B_{UL}, the stimulated emission coefficient; and B_{LU}, for absorption. The quantities η_{UL}, N_{UL} and ϕ_{LU} are the line shape functions for the three respective processes, which are generally different.

Characteristic times for the spontaneous decay of low-lying vibrational states for molecular species of interest are of the order 10^{-1} to 10^{-3} second, whereas other rate processes are typically much faster. Hence, in the equation above, the spontaneous emission term generally can be neglected. Also, for the present analyses, interest focuses primarily on photon processes occurring at line center. At line center $\phi_{LU} = \eta_{UL}$. Thus,

$$\frac{dI\nu_\sigma}{dS} = h\nu\phi_{LU}(\nu_\sigma) \left[B_{UL}\ n_u - B_{LU} n_L\right] I_{\nu o} \tag{115}$$

The factor multiplying $I_{\nu o}$ is the optical gain coefficient, viz:

$$g_{UL} = h\nu\phi_{LU}\nu_o \left[B_{UL}\ n_u - B_{LU} n_L\right] \tag{116}$$

The Einstein coefficients are connected by the relationship

$$\frac{B_{LU}}{B_{UL}} = \frac{d_u}{d_L} \tag{117}$$

where d_U and d_L are degeneracies (statistical weights) of the upper and lower states, respectively. Also, it is possible to write

$$B_{LU} = \frac{8\pi^3}{3h^2c} \left| R_{LU} \right|^2 \tag{118}$$

where R_{LU} is the quantum-mechanically-derived transition matrix element. Hence, the gain expression may be rewritten as

$$g_{UL} = \frac{8\pi^3}{3h} \left(\frac{\nu o}{c}\right) \phi_{LU} \left(\nu_o\right) \left| R_{LU} \right|^2 \left[\frac{n_u}{d_u} - \frac{n_L}{d_L}\right] \qquad (119)$$

or

$$g_{vj}^{v'j'} = \frac{8\pi^3}{3h} \frac{\nu o}{c} \phi_{LU} \nu_o \left|R_{LU}\right|^2 \left[\frac{n_{vj}}{d_{vj}} - \frac{n_{v'j'}}{d_{v'j'}}\right] \qquad (120)$$

Consider vibrational-rotational transitions of the form

$$(v+1, J) \rightleftarrows (v,J)$$

where v is the vibrational quantum number and J is the rotational quantum number.

Then

$$\left|R_{LU}\right|^2 = S_j \left|R_{v,\ v+1}\right|^2 \qquad (121)$$

where:

$$S_J = \begin{matrix} J \text{ for P-branch transitions (i.e., } J' = J + 1) \\ J + 1 \text{ for R-branch transitions (i.e., } J' = J - 1) \end{matrix}$$

$R_{v',\ v+1}$ = vibrational-transition matrix element

At pressures of a few torr or less, transitions are predominately Doppler broadened. At higher pressures, the combined influence of Doppler and pressure (Lorentz) broadening is present. Therefore, the line-shape factor $\emptyset_{LU}$ (ν_o) is represented in terms of a Voight profile such that

$$\frac{\nu_o}{C}\,\phi_{Lu}(\nu_o) = \left(\frac{m}{2\pi KT}\right)^{\frac{1}{2}} \exp(\xi)^2 \,\mathrm{erfe}\,(\xi) = \left(\frac{m}{2\pi KT}\right)^{\frac{1}{2}} \phi(\xi) \tag{122}$$

$$a_D:(\nu_o) = \left(\frac{K}{n}\right)\left(\theta_{001} - \underset{(\theta_{02^\circ 0})}{\theta_{001}} + J\,(J+1)\,\theta^{001}_{rot} - J'\,(J'+1)\,\underset{(\theta^{020}_{rot})}{\theta^{100}_{rot}}\right) \tag{123}$$

$$\sigma_{CO_2} - CO_2 = \frac{001 \rightarrow 100}{10.5 \times 10^{-15}\ cm^2} \tag{124}$$

$$\sigma_{CO_2} - CO_2 = \frac{001 \rightarrow 02^\circ 0}{10.2 \times 10^{-15}\ cm^2} \tag{125}$$

The influences of Doppler broadening and vibration-rotation interaction have been taken into account.

where

$$\xi = \frac{a_p}{a_D}\sqrt{\ln 2} \tag{126}$$

a_p = pressure broadened (Lorentz) half-width

$$= \frac{n}{2\pi C} \sum_s V_s X_s \sigma_s$$

a_D = Doppler broadened half-width

$$= \frac{\gamma_o}{C^2}\sqrt{\frac{2kT(\ln 2)}{m}}$$

v_s is the mean relative velocity $(\sqrt{2kT/M})$ between the emitting molecule and the colliding species; X_s is the species mole fraction, σ_s is the broadening cross-section due to the impacting species s; ν_o is the transition frequency at line center; m is the mass of the emitter molecule; and M is the reduced mass between an emitter molecule and the collider molecule of species s:

$$\mu = \frac{mm_s}{m + m_s} \tag{127}$$

The optical gain coefficient may be rewritten as

$$g(V,J) = \frac{8\pi^3}{3h} \left(\frac{m}{2\pi kT}\right)^{\frac{1}{2}} \phi(\xi) S_J \left|R_{V,V+1}\right|^2 \left[\frac{n_{V+1,J'}}{d_J} - \frac{n_{V,J}}{d_J}\right] \tag{128}$$

Here the quantities V and J in the expression g(V,J) indicate the lower levels of the transition.

In treating the populations of the vibrational-rotational levels, it is assumed that the rotational mode can be described by the local translational temperature T. Hence,

$$n_{V,J} = \left(\frac{n_{V,J}}{n_V}\right) n_V = \frac{d_J \exp\left[-1.439\, J(J+1)(B_e - \alpha_e\, (V+\frac{1}{2})/T\right] n_V}{Q_{rot}\ (V)} \tag{129}$$

where B_e is the spectroscopic rotational constant (cm^{-1}), and α_e is its anharmonic correction. The quantity $Q_{rot}^{(V)}$ is the rotational partition function, which is evaluated according to the relation

$$Q_{rot}\ (V) = \sum_J (2J+1) \exp\ (-E_{rot}\ (J,V)/kT) \tag{130}$$

The populations can also be represented by:

$$n_V J = n_V f_J = n_V \left[\frac{2J+1}{Q_{rot}^{(V)}}\right] \varepsilon^{\left(\frac{-J(J+1)}{kT}\ \theta_{rot}^{(V)}\right)} \tag{131}$$

where,

$$Q_{rot}^{(V)} = \frac{T}{2\theta_{rot}^{(V)}}$$

$$\frac{n_{VJ}}{g_{VJ}} = \frac{nV}{gV} \exp\left(\frac{-J(J+1)}{kT}\right) \theta_{rot}^{(V)}$$

$$\frac{n_V}{g_V} = n_{000} \exp\left(-\theta_V / T_V\right)$$

θ_V = characteristic temp. of state

T_V = vibrational temperature of state

g_V, g_{VS} represent degeneracies

For the transitions

$$CO_2\ (001,J) \longrightarrow CO_2\ (100,J\pm1)$$

$$CO_2\ (001,J) \longrightarrow CO_2\ (02^\circ0, J\pm1)$$

the pertinent constants are:

$$R_{001,100} = 0.0331 \times 10^{-18} \text{ esu-cm}$$

$$R_{001,02^\circ0} = 0.0295 \times 10^{-18} \text{ esu-cm}$$

$$\theta_{rot}^{(001)} = 0.55632 \quad K$$

$$\theta_{rot}^{(02^\circ0)} = 0.56106 \quad K$$

$$\theta_{rot}^{(100)} = 0.56078 \quad K$$

$$\theta_{001} = 3380 \quad K \qquad \theta_{100} = 1997 \quad K$$

$$\theta_{020} = 1850 \quad K$$

The expressions for the gain coefficients on two transitions are

$$g_{001,J}^{700,J} = (0.79 \text{ X } 10^{-14})|m|(1 - 0.0044m)T^{-\frac{3}{2}} n X_{000} \phi \left[(0.55632) \exp\left(\frac{-3380}{T_{001}} - J(J+1)\ (0.55632/T)\right) - (0.56078) \exp\left(\frac{-2000}{T_{100}} - J'(J'+1)\ (0.56078/T)\right)\right] \tag{132}$$

$$g_{001,J}^{020,J'} = (0.63 \text{ X } 10^{-14})|m|(1-0.006m)T^{-\frac{3}{2}} n X_{000} \phi \left[(0.55632) \exp\left(\frac{-3380}{T_{001}} - J(J+1)\ (0.55632/T)\right) -(0.56106) \exp\left(\frac{-1850}{T_{020}} - J'(J'+1)\ (0.56106/T)\right)\right] \tag{133}$$

where $m = -(j + 1)$ $J' = J + 1$ (P)

$m = J$ $J' = J - 1$ (R)

n = total number density $= \frac{P}{KT}$

X_{000} = mole fraction of ground state CO_2 (from program)

$J' = 0,2,4,6, \ldots\ldots\ldots\ldots$

For largely pressure-broadened line, Ø may be expressed as:

$$\phi \approx \frac{1}{\sqrt{\pi}\,\xi} \left[1 - \frac{0.5}{\xi^2} + \frac{0.75}{\xi^4} - \frac{1.875}{\xi^6} + \frac{6.5625}{\xi^8} - \ldots\right] \tag{134}$$

Argument List

XIC	The field (matrix) of individual intensities in the calculation region
GAN	Gain (updated) of each of the point locations of the field
IXMAX	Number of points in the flow direction

DXCAV	The distance between points in the x-direction
IFIELD	Indicator for small signal gain (IFIELD = 1) or Loaded Gain (IFIELD ≠ 1)
IY	Number of flow streams, i.e., points in the y-dimension.

Commons Modified

	/PROPT/
TS	Static temperature in the cavity (K)
PS	Static pressure in the cavity
V	Gas velocity (cm/sec)
RHO	Gas density (gm/cc)
RHON	Number density (particles/cc)
	/ENERG/
EN2	Energy (population) of the V = 1 level of N_2
EOOV	Energy (population) of the asymmetric stretch vibration mode
EOVO	Energy (population) of the bending vibration mode of CO_2
EVOO	Energy (population) of the symmetric stretch mode of CO_2
	/RATE/
RSTIM	Rate for stimulated emission.

SUBROUTINE KINET   76/176   OPT=1   FIN 4.6+452   04/27/79 12.23.47

```
      SUBROUTINE KINET(XIC,GAN,IAMAX,DXCAV,IFIELD,IY)
C     CO2 KINETICS ROUTINE
C     THIS ROUTINE CALCULATES GDL GAIN (10.6) AS A FUNCTION OF KINETIC
C     AND STIMULATED EMISSION EFFECTS.
      LEVEL 2, XIC,GAN
      COMMON/PROPT/TS,PS,V,RHO,RHON,CP,GAMMA,R,B,XLAMB,HNU,CPRM
      COMMON/START/TSI,PSI,VI,EOOVI,EOVOI,EVOOI,EN2I,GAINI
      COMMON/MOLES/XN2,XCO2,XH2O,XCO,XO2
      COMMON/ENERG/EN2,EOOV,EOVO,EVOO
      COMMON/RATE/RN2,RC3,RC2,RPUMP,     RSTIM
      COMMON/FACTER/XMW,AG,GCON,ROTUP,ROTLO,RCORR,C
      DIMENSION GAN(  1   ),XIC(  1   ),SOEV(190)
      IF(IFIELD .EQ. 1) IY=1
C     IF(IFIELD .EQ. 1) CALL ZERO(XIC( 1 ),XIC(16384))
      IF (IFIELD.NE.1) GO TO 174
      DO 173 IZERO=1,16384
  173 XIC(IZERO) = 0.
  174 F3 = 2.349E10/HNU
      F4 = 1.388E10/HNU
      F5 = GAMMA*R
      F6 = XMW/AG
      F7 = XCO2*2349.
      DO 200 J=1,IY
      TS = TSI
      PS = PSI
      V = VI
      GAIN = GAINI
      RHO = PS/R/TS*1.013E6
      RHON = RHO/XMW*AG
      T2 = 959.8 / ALOG(1.+XCO2*1334./EOVOI)
```

```
      EGL = EOVOI + EVOOI
      EN2 = EN2I
      EVOO = EVOOI
      EOVO = EOVOI
      EOOV = EOOVI
      X = 0.0
      SUMDEV = 0.0
      IBAR = 0
      XCAV = 0.0
   10 IBAR = IBAR+1
      XCAV = DXCAV*(IBAR-1)+DXCAV/2.
      IF(XCAV.LT.X) GO TO 100
      CALL MIX
   20 G1 = GAIN
      F1 = EXP(3354./TS)
      F2 = EXP(3380./TS)
      IF(IBAR.EQ.1) GO TO 6
      IJ = (X+DXCAV/2.)/DXCAV
      IP = IJ+(J-1)*IXMAX
      XI =XIC( IP )+(XIC(  IP+1)-XIC( IP ))/DXCAV*(X-IJ*DXCAV+(DXCAV/2.)
     X)
      GO TO 7
    6 XI = XIC(1+(J-1)*IXMAX)*X/(DXCAV/2.)
    7 CONTINUE
      SUM1 = SUMDEV
      DT = 1./(7.0*AMAX1(RC2,RPUMP,RSTIM))
      EQN2 = XN2/(F1            -1.)*2331.
      EQOOV = XCO2*2349./(F2           -1.)
      EQOVO = XCO2*1334./(EXP(959.8/TS)-1.)
      XA = 1.-EN2/EQN2
      XB = 1.-EOOV/EQOOV
      EPSL = -25.9/TS
      YA = 1.-1./F1
      YB = 1.-1./F2
      XAB = 1./YA*(XA-XB-(EPSL*XA*(XB-1.))/(F1             -1.))
      XADOT = -YA*XCO2*XAB*RPUMP
      XBDOT = YB*XN2*EXP(-EPSL)*XAB*RPUMP
      DEN2MP = (EN2-EQN2)*RN2*DT
      DEN2 = EQN2*XADOT*DT + DEN2MP
      F10 = XI*GAIN/RMON
      DEOOVR = (EOOV-EQOOV)*RC3*DT
      DEOOV = DEOOVR +(F3*F10+EQOOV*XBDOT)*DT
      DEOVO = (EOVO-EQOVO)*RC2*DT
      DEGL = DEOVO-1.094*DEN2MP-1.086*DEOOVR-F4*F10*DT
      EN2 = EN2-DEN2
      EOOV = EOOV-DEOOV
      EGL = EGL-DEGL
      SUMDEV = SUMDEV + DEOVO*V*1.987E-16*RMON
      DX = V*DT
      X = X + DX
      PS = PS*1.013E06
      DEV = DEVO/DT*1.19677E8
      Q = V*V/(F5      *TS)-1.0
      PP = DEV/CP/TS
      V = V-PP/Q*V*DT
      RMO = RMO+PP/Q*RMO*DT
      RMON = RMO/F6
      PS = PS+PS*PP*GAMMA*(Q+1.)/Q*DT
      TS = PS/RMO/R
      PS = PS/1.013E6
      CHI2 = -959.8/T2
      Y= CHI2
      Z1 = EXP(77.71/TS)
   31 F8 = EXP(-Y)
      F9 = EXP(-2.*Y+77.71/TS)
      FX = EGL-XCO2*(1388./(F8*F8*Z1-1.)+1334./(F8-1.))
      FP1 = XCO2*(2776.*F9/(F9-1.)**2+1334.*F8/(F8-1.)**2)
      FPX = -FP1
      YOLD = Y
      Y = YOLD - FX/FPX
      IF (ABS((Y-YOLD) / Y).GT. 1.E-3) GO TO 31
      T2 = -959.8          / Y
```

```
      T1 = 1388./(1334./T2 + 54./TS)
      EV00 = 1388.*XCO2/(EXP(1997. /T1   ) - 1.)
      E0V0 = XCO2*1334./(EXP(959.8      /T2)-1.)
      CHI2 = Y
      CHI1 = 2. * CHI2 - 77.71         / TS
      Q1 = 1./(1.-EXP(CHI1))
      Q2 = 1./(1.-EXP(CHI2))
      Q3 = E00V/F7+1.
      T3 = -3380./ALOG(1.-1./Q3)
      X000 = XCO2/(Q1*Q2*Q2*Q3)
      APAD = CPRM*RMON
      WORM = .8326*APAD
      IF(WORM.GT.10.) GO TO 40
      PHI = EXP(WORM**2)*ERFC(WORM)
      GO TO 41
   40 PHI = 0.67764/APAD
   41 CONTINUE
      TFACT = TS**(-1.5)
      GAIN = GCON*TFACT*RMON*X000*PHI*(.556*EXP(-3380./T3-HOTUP/TS)
     X  -.561*EXP(-1997./T1-HOTLO/TS))
      BIGSIG = GCON*TFACT*PHI*EXP(-HOTUP/TS)*.556
      RSTIM = XI*BIGSIG/HNU*1.E7
      IF(X.LE.XCAV) GO TO 20
  100 GAN(IBAR+(J-1)*IXMAX ) = GAIN-(GAIN-G1)*(X-XCAV)/DX
      SDEV(IBAR)     = SUMDEV-(SUM1-SUMDEV)*(X-XCAV)/DX
      IF(IBAR.EQ.IXMAX) GO TO 300
      GO TO 10
  300 DO 301 I = 1,IXMAX
  301 XIC(I+(J-1)*IXMAX ) = SDEV(I)
  200 CONTINUE
C     DO 60 J=1,IY
C     WRITE(6,205)(XIC(I,J),I=1,IXMAX)
C     WRITE(6,203)(GAN(I,J),I=1,IXMAX)
C  60 WRITE(6,204)(SDEV(I)   ,I=1,IXMAX)
C     WRITE(6,201)X,EN2,E00V,EGL
C     WRITE(6,202)TS,PS,V,RHO,Q
C 201 FORMAT(5X,24H--KINET-- X,EN2,E00V,EGL,5X,4E15.5/)
C 202 FORMAT(10X,24H--KINET--  TS,PS,V,RHO,Q,5X,5E15.5/)
C 203 FORMAT(10X,13H--KINET-- GAN/5(10E12.4/))
C 204 FORMAT(10X,14H--KINET-- SDEV/5(10E12.4/))
C 205 FORMAT(10X,19H--FIELD INTENSITY--/5(10E12.4/))
      RETURN
      END
```

17. SUBROUTINE MIRROR

a. Purpose -- MIRROR applies a mirror transmission function to the complex field which may include reflectivity, clipping, radius of curvature, edge diffraction imaging, small tilt, astigmatism, localized surface distortion, and overall spherical distortion. In addition, two specialized options have been included: (1) a toric mirror effect for axicon optics and (2) a mirror dimple effect which enables a localized difference in radius of curvature. Figure 38 shows the subroutine MIRROR organization. Computer printouts of the MIRROR subroutine begin on page 168.

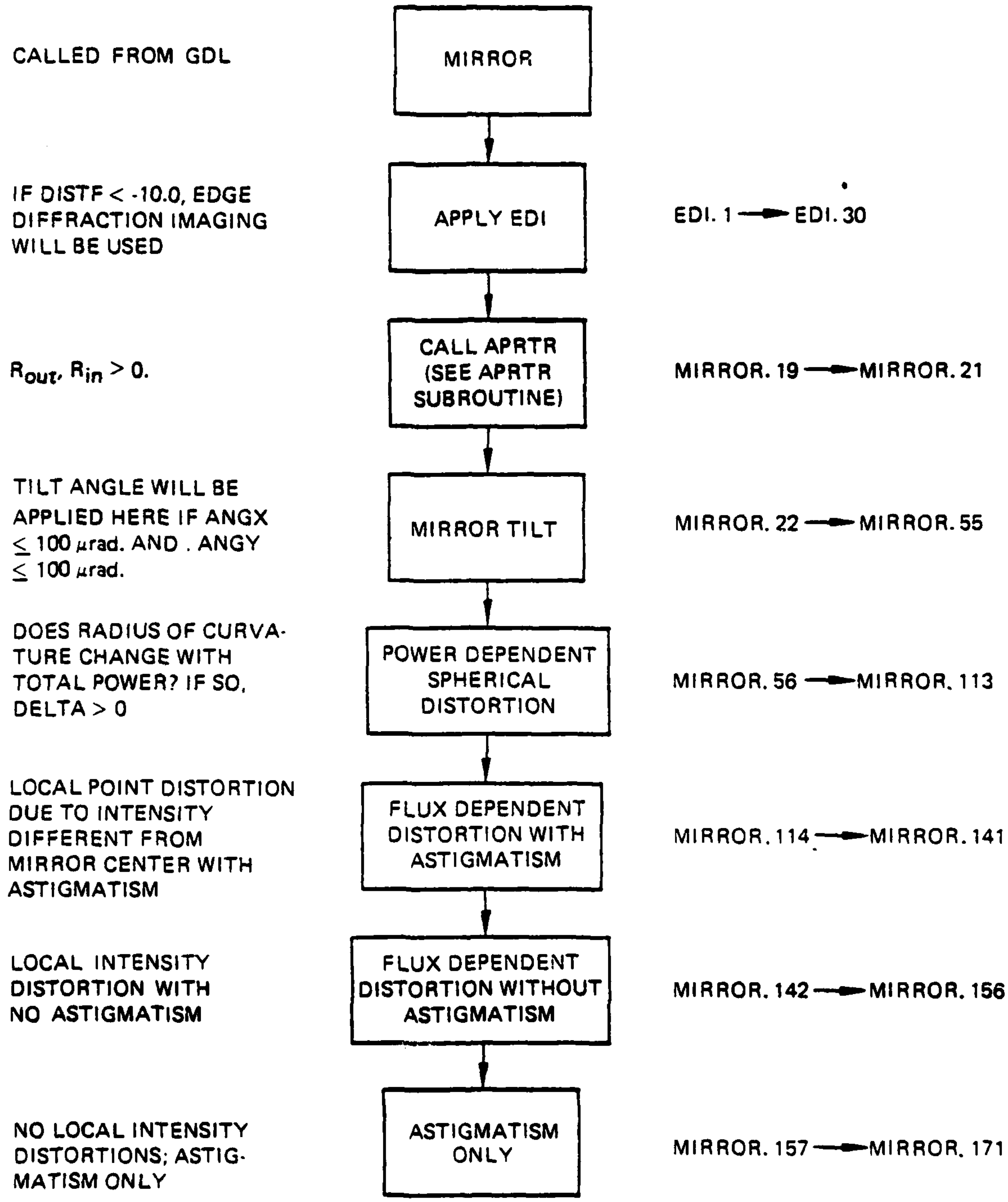


Figure 38. Subroutine MIRROR organization.

The routine first tests for the option of edge diffraction imaging in which the outer annular edge of the mirror has a radius of curvature different from the mirror. When this option is used the MIRROR subroutine must be called separately to apply EDI.

The subroutine must be called again for the rest of the mirror.

The routine then apertures the field to the size of the mirror and applies small mirror misalignments (angles less than 100 microradians) to the field. For large angles, the angle information is stored in ANGX and ANGY which are located in common MRPROP and used to later determine the location of the center of the field. The field itself is not altered for the large angles.

b. Revelant formalism -- A distortion-free mirror is applied to the field in Figure 39 by changing the optical path lengths of the field points. For example, apply a convex mirror to a plane wave.

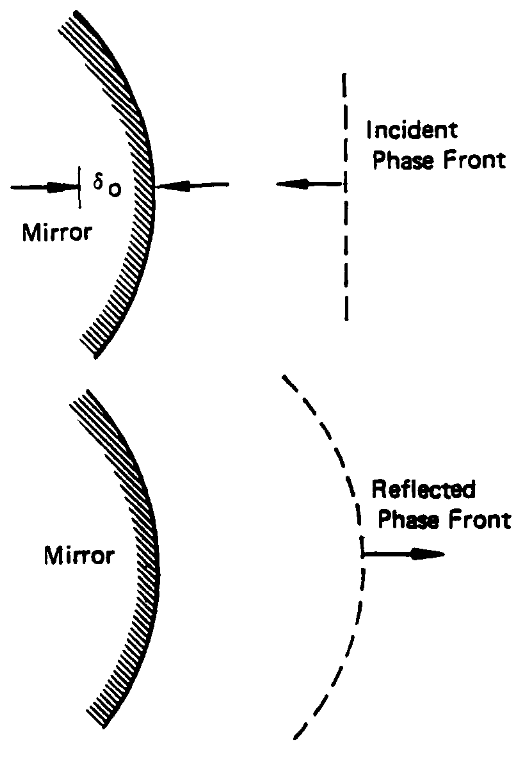


Figure 39. Mirror transmission function relative to the complex field.

Note that the field at the edge has traveled $2\delta_o$ more than the center. The size of the sag $\delta(r)$ (Fig. 40) at any point r can be found from the sag formula:

$$(R_c - \delta)^2 + r^2 = R_c{}^2 \tag{135}$$

$$\delta \quad \frac{r^2}{2R_c} = \frac{x^2 + y^2}{2R_c}$$

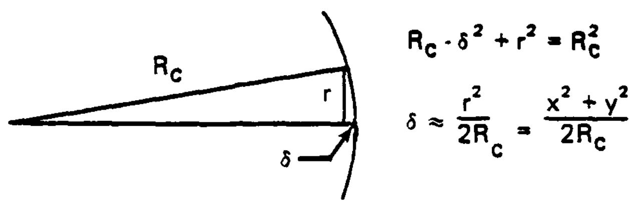


Figure 40. Graphic representation of SAG.

Thus, to make the center of the field lead the edge by a factor of $2\delta_o$, the following transmission function is applied to the field:

$$u'(x,y) = T(x,y)\, u(x,y), \quad T(x,y) = e^{\frac{2\pi}{\lambda} i \left(\frac{x^2 + y^2}{R_c}\right)} \tag{136}$$

The sign convention used is a negative radius of curvature for a convex mirror. A concave mirror has a positive radius of curvature.

In addition to curvature, the MIRROR routine can apply power or flux dependent distortions to the field.

The power dependent mirror distortion can be applied given the center-to-edge maximum sag, DELTA, determined by design power, PWRDES. The incident power is then calculated and the sag reduced by the ratio of incident power to design power. For a ratio greater than one, it is assumed that the sag is that of the design power.

The flux dependence is applied assuming a distortion factor, DISTF, which weights intensity changes from the center of the field and thus applies an intensity-dependent phase factor to the field.

Astigmatism can be applied to the field in conjunction with the localized flux-dependent distortion or can be applied alone. Astigmatism is included if PHIAST is input (as a number greater than 0). PHIAST is the angle between the mirror normal and the optical axis (in degrees). The phase is altered by astigmatism by computing separate (sagittal and tangential) radii of curvature for the mirror and applying to vary the X and Y component of the phase field, respectively.

Argument List

ANX	Mirror tilt in X (about y-axis)
ANY	Mirror tilt in Y (about x-axis)
RADC	Radius of curvature of mirror (cm)
RIAOUT	Outside radius (cm)
RIAIN	Inside radius of annular mirror (cm)
XPOS	X-direction offset of mirror centerline from optical axis of beam (cm)
YPOS	Y-direction offset of mirror centerline from optical axis of beam (cm)
RFL	Mirror reflectivity - fraction $0.0 \rightarrow 1.0$
DELTM	Total power spherical distortion factor
DISTF	Flux distortion factor - local intensity distortion $f(I_{local} - I_{center})$; $(cm/W/cm^2)$
RANULS	Radius to annulus for toric mirror option
RYOUT	Outside Y-dimension (from center) for a rectangular mirror (cm)
RYIN	Inside Y-dimension (from center) for a rectangular mirror (cm)
PHIAST	Angle of beam with respect to mirror normal (deg)

Relevant Variables

AKY	$2\pi/WL = 2\pi/\lambda$ where $\pi = 3.14159$
ANGX	Accumulative x-dim angle to trace field in cavity
ANGY	Accumulative y-dim angle to trace field in cavity

COSP	Cosine of phase change
CUR	Real array representing the complete wave amplitude field, i.e., Intensity (J) = $\left[CUR(J)\right]^2 + \left[CUR(J-1)\right]^2$
DELTA	DELTM, total power spherical distortion factor
EMF	Square root of mirror reflectivity
PHASE	Phase change at each point of wavefront
PHI	Phase change in TORIC MIRROR and DIMPLE calculations
PPW	Integrated power on mirror
RADCUR	Negative focal length of mirror (-f)
RMSAG	Sagittal radius of curvature (astigmatism)
RMTAN	Tangential radius of curvature (astigmatism)
SINP	Sine of the phase change
WL	Wavelength, λ
WNDW	Magnification factor for scaling power
XX	X^2; x-component of location, squared
YY	Y^2; y-component of location, squared

Commons Modified

/MELT/

Array modified CUR(I) @ MIRROR 50, 51, 98, 99, 139, 140, 167

/MRPROP/

Variables modified:	RADCUR	@	MIRROR 115
	ANGX	@	MIRROR 25
	ANGY	@	MIRROR 26

EDI (Edge Diffraction Imaging)

Power near the outer edge of the beam that would have been ordinarily lost through diffraction is partially recovered by incorporating a separate radius of curvature in an outer edge annulus, as shown in Equation 137 and Figure 41.

$$\Delta\phi = W_n \left[(x^2+y^2) - R_{in}^{\;2}\right]/R_{EDI} \qquad W_n = \frac{2\pi}{\lambda} \tag{137}$$

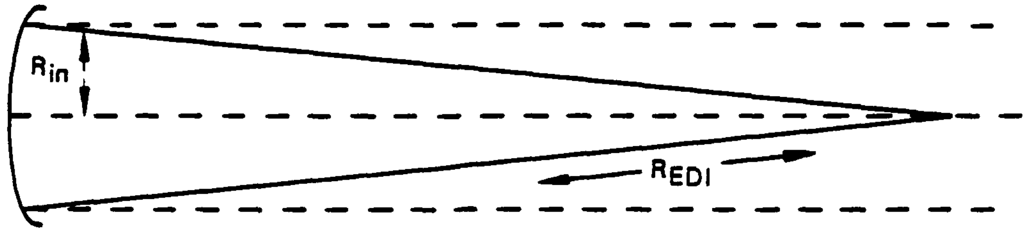


Figure 41. Edge diffraction imaging.

The real representation CUR of the complex amplitude field is modified as follows:

K2 = EVEN NOS

K2MI = ODD NOS

Real Part: CUR (K2) = CUR (K2-1) (sin Ø) + CUR (K2) (cos Ø)

Im part: CUR (K2M1) = CUR (K2-1) (cos Ø) - CUR (K2) (sin Ø)

This array is modified in the same way by the phase changes throughout this subroutine.

Mirror Tilt (<100 μrad)

$$\Delta\phi = -2 \left[(ANX)(X) + (ANY)(Y)\right]\frac{2\pi}{\lambda} \tag{138}$$

where

ANX => tilt in x-direction ∿ radians

ANY => tilt in y-direction ∿ radians

Power Dependent Spherical Distortion

This part of MIRROR subroutine calculates the phase change due to total power induced spherical distortion.

$$\Delta\phi = \frac{2\pi}{\lambda} \left[\frac{(x^2+y^2)}{R}\right] \tag{139}$$

where

$$R = R^2_{out}/2\delta$$

and δ = DELTA = MAX (Center) DISTORTION $\frac{\text{(Scaled}}{\text{for Power)}}$

$$DELTA = DELTA^{(1)} \left(\frac{P_{incident}}{P_{design}}\right)$$

R_{out} = Center to edge mirror radius

(1) this is the input DELTA=DELTM

Flux Dependent Distortion (No Astigmatism)

Flux Dependent Distortion + Astigmatism

$$\Delta\phi = \frac{2\pi}{\lambda}\left[\frac{x^2}{R_{SAG}} + \frac{y^2}{R_{TAN}}\right] - \frac{2\pi}{\lambda}\,\delta_I\,(1\text{-Ref.})\; 2\,\frac{I_{CL} - I_{xy}}{(WNOW)^2} \tag{140}$$

where

$$R_{SAG} = RADC/\cos\phi_{ast}$$

and

$$R_{TAN} = RADC\,(\cos\phi_{ast})$$

where

RADC = radius of curvature of mirror (cm)

ϕ_{AST} = beam-mirror angle radians = PHIAST $\frac{\pi}{180}$

I_{CL} = Mirror centerline intensity

I_{XY} = Local (X,Y) intensity

WNOW = VAMP power correction factor

δI = Flux distortion factor (cm/W/cm^2)

Ref. = Mirror reflectivity

$$\Delta\phi = \frac{-2\pi}{\lambda}\,\delta_I\,(1\text{-Ref})\; 2\,\frac{I_{CL} - I_{xy}}{(WNOW)^2} \tag{141}$$

where,

δ_I = Flux distortion factor (cm/W/cm^2)

I_{CL} = Intensity at mirror center (W/cm^2)

I_{xy} = Intensity at coordinated x,y (W/cm^2)

Ref = Mirror reflectivity

WNOW = VAMP power correction factor

Astigmatism (Only)

$$\Delta\phi = \frac{2\pi}{\lambda}\left[\frac{x^2}{R_{SAG}} + \frac{y^2}{R_{TAN}}\right] \tag{142}$$

where R_{SAG} = RADC/cos ϕ ast (cm)

and R_{TAN} = RADC (cos ϕ ast) (cm)

where RADC = radius of curvature of mirror (cm)

ϕ_{ast} = beam - mirror (astigmatic) angle

= PHIAST $\left(\frac{\pi}{180}\right)$

SUBROUTINE MIRROR 76/176 OPT=1 FIN 4.6+452 04/27/79 12.23.47

```
      SUBROUTINE MIRROR(ANX,ANY,RADC,RIAOUT,RIAIN,XPOS,YPOS,RFL,DELTM,   MIRROR     2
     X DISTF,RANDLS,RYOUT,RYIN,PHIAST)                                   CIDASTG    8
C     MODIFIED BY JCC 11/4/75 TO MAKE COMPLEX MULTIPLY MORE EFFICIENT.   MIRROR     4
C      MIRROR TRANSMISSION FUNCTION                                      MIRROR     5
C      THIS ROUTINE APPLIES A MIRROR TRANSMISSION FUNCTION TO THE        MIRROR     6
C      COMPLEX FIELD.  THE FOLLOWING EFFECTS ARE INCLUDED:               MIRROR     7
C       1. EDGE AND CENTRAL OBSCURATIONS                                 MIRROR     8
C       2. MIRROR RADIUS OF CURVATURE                                    MIRROR     9
C       3. MIRROR REFLECTIVITY                                           MIRROR    10
C       4. POWER DEPENDENT DISTORTION                                    MIRROR    11
C       5. FLUX DEPENDENT DISTORTION                                     MIRROR    12
C       6. TORIC MIRROR RADIUS OF CURVATURE                              MIRROR    13
      LEVEL 2, CUR                                                       MIRROR    14
      COMMON/MELT/CUR(32768),CFIL(16512),X(128),WL,NPTS,NPY,DRX,DRY      MIRROR    15
      COMMON/MRPROP/RADCUR,ANGX,ANGY                                     MIRROR    16
      COMMON/WAY/WNOW,NREG,RAPTR                                         MIRROR    17
      COMPLEX CFIL                                                       CONR2      8
      IF(RIAOUT.EQ.0.0.AND.RIAIN.EQ.0.0) GO TO 70                        MIRROR    19
      IF (DISTF.LE.-10.) GO TO 300                                       EDI        1
      CALL APRTR(RIAOUT,RIAIN,XPOS,YPOS,RYOUT,RYIN)                      SDAPR     38
C                        ********************                            MIRROR    21
C *** MIRROR TILT ADDITION THROUGH STATEMENT NO 50          ***********  MIRROR    22
C                        ********************                            MIRROR    23
   70 IF (ABS(ANX) .LE. .000100 .AND. ABS(ANY) .LE. .000100) GO TO 71    MIRROR    24
      ANGX=ANX*2.0  +  ANGX                                              MIRROR    25
      ANGY=ANY*2.0  +  ANGY                                              MIRROR    26
   71 DELTA=DELTM                                                        MIRROR    27
      FMF=SQRT(RFL)                                                      MIRROR    28
      AKY = 2.0 * 3.14159 / WL                                           MIRROR    29
      PI = 3.14159                                                       MIRROR    30
      PWRFAC = 0.                                                        MIRROR    31
```

```
      DISMAX =  100000.
      PPW =0.
      PMLT1 = -100000.
      PWRDES = 500000.
      IF (RANULS .GT. 0.0)GO TO 100
      IF (RIAOUT .LT. 0.) GO TO 200
      IF (ABS(ANX) .GE..000101 .OR. ABS(ANY) .GE. .000101) GO TO 50
      IF (ANX .EQ. 0. .AND. ANY .EQ. 0.0) GO TO 50
      DO 60 J = 1,NPY
      J1= (J-1) * NPTS
      DO 60 I = 1,NPTS
      TILT  = ANX * X(I) +  ANY * X(J)
      PHASE = -2.0 * TILT * AKY
      K2 = 2 * ( I + J1 )
      K2M1 = K2 - 1
      SINP = SIN (PHASE)
      COSP = COS (PHASE)
      CURS = CUR(K2M1)
      CUR(K2M1) = CURS*COSP - CUR(K2)*SINP
   60 CUR(K2)    = CURS*SINP + CUR(K2)*COSP
   50 NOB = NPTS * NPY
      DELMAX = 0.
      DELIN = 0.
C****                                                          ******
C ******* POWER DEPENDENT RADIUS OF CURVATURE CALCULATIONS ARE  ******
C ****  ON .... PHASE = 2 PI/LAMDA(X**2 +Y**2/2 R)..........   ******
C ****   R =F(DESIGN POWER, INCIDENT POWER, CENTER TO EDGE DISTORTION***
C ****   WHERE DESIGN POWER = PWRDES                            *****
C ****          INCIDENT POWER = PPW                            *****
C ****          MAX C. TO  E. DISTORTION = DELTA                *****
C ****              J FORGHAM  11 - 15  - 74                    *****
C ******************************************************************
      DELIN = DELTA
      IF (DELTA .EQ. 0. .OR. FMF .EQ. 1.) GO TO 30
      IF (DELTA .LT. 0.) GO TO 20
      DELMAX = ABS(DELTA)
C ****************             *****            ***************
C****** POWER   SCHEDULED   CENTER   TO   EDGE   DISTORTION *********
C*******************           *****             *************
      DO 15 I=1, NOB
      I2 = 2 * I
   15 PPW = PPW + CUR(I2-1)**2 + CUR(I2)**2
      PPW = PPW * (X(2) - X(1))**2 * (NPTS/NPY)
      IF (NREG .EQ. 1 .OR. NREG .EQ. 2 ) PPW = PPW/WNOW**2
C     PWRFAC = POWER IN BEAM /DESIGN POWER
      PWRFAC = PPW /PWRDES
      IF (PWRFAC .GT. 1.) PWRFAC =1.
      DELTA = PWRFAC * DELTA
      GO TO 21
   20 DELTA = ABS(DELTA)
   21 RADIUS =(RIAOUT**2) * PI/(WL * AKY * DELTA)
      ROC = -RADIUS
      ETA = AKY /ROC
      RSQ = RIAOUT**2
      DO 23 I = 1, NPY
      YSQ = X(I)**2
      DO 23 J = 1, NPTS
      ARG = YSQ + X(J)**2
      IF (ARG .GT. RSQ) GO TO 23
      PHIMIR = ETA * ( YSQ + X(J)**2 )
      IJ = J + ( I - 1) * NPTS
      IJ2 = IJ * 2
      IJ2M1 = IJ2 - 1
      SINP = SIN(PHIMIR)
      COSP = COS(PHIMIR)
      CURS = CUR(IJ2M1)
      CUR(IJ2M1) = CURS*COSP - CUR(IJ2)*SINP
      CUR(IJ2)    = CURS*SINP + CUR(IJ2)*COSP
   23 CONTINUE
      IF (DELIN .LT. 0.) GO TO 25
      WRITE(6,89)
   89 FORMAT(///,20X,35HPOWER SCHEDULED MIRROR DISTORTION
      WRITE (6,90)ROC, DELMAX, DELTA, PWRFAC
```

```
90 FORMAT(//,39H  POWER INDUCED RADIUS OF CURVATURE =  ,G12.5,2HCM,/,   MIRROR     105
  X39H  MAXIMUM CENTER TO EDGE DISTORTION =  ,G12.5,2HCM,/,             MIRROR     106
  X39H  APPLIED CENTER TO EDGE DISTORTION =  ,G12.5,2HCM,/,             MIRROR     107
  X39H  FRACTION OF DESIGN POWER INCIDENT =  ,G12.5,//)                 MIRROR     108
   GO TO 30                                                             MIRROR     109
25 WRITE (6,91)DELTA,ROC                                                MIRROR     110
91 FORMAT(//,20X,18HMIRROR DISTORTION ,//,                              MIRROR     111
  X37H APPLIED CENTER TO EDGE DISTORTION = ,G12.5,2HCM,/,               MIRROR     112
  X42H DISTORTION INDUCED RADIUS OF CURVATURE = ,G12.5,2HCM)            MIRROR     113
30 IF(ABS(RADC).GT.0.)AKYR=AKY/RADC                                     MIRROR     114
    RADCUR=-RADC/2.                                                     MIRROR     115
    IF(PHIAST .EQ. 0.0) GO TO 350                                       C10ASTG      9
    PHIR =( PHIAST * PI)/180.                                           C10ASTG     10
    RMSAG = RADC / COS(PHIR)                                            C10ASTG     11
    RMTAN = RADC * COS(PHIR)                                            C10ASTG     12
350 CONTINUE                                                            C10ASTG     13
    NP=NPTS/2                                                           MIRROR     116
    NDEX=(NP-1)*NPTS+NP                                                 MIRROR     117
    IF (FMF.EQ.1.0.AND.DISTF.EQ.0.0)GO TO 10                            MIRROR     118
    ALPHA = 1.0-FMF**2                                                  MIRROR     119
    DELF=DISTF*ALPHA                                                    MIRROR     120
    DELF2=DELF*2.                                                       MIRROR     121
    XICL=CUR(2*NDEX-1)**2 + CUR(2*NDEX)**2                              MIRROR     122
    IF (ABS(RADCUR).LT..5) GO TO 2                                      MIRROR     123
    WNOWSQ=1.0                                                          MIRROR     124
    IF (NREG .EQ. 1 .OR. NREG .EQ. 2 ) WNOWSQ=WNOW**2                   MIRROR     125
    JJ = 0                                                              MIRROR     126
    DO 1 I=1,NPY                                                        MIRROR     127
    YSQ = X(I)**2                                                       MIRROR     128
    DO 1 J=1,NPTS                                                       MIRROR     129
    JJ = JJ + 1                                                         MIRROR     130
    JJ2 = JJ * 2                                                        MIRROR     131
    JJ2M1 = JJ2 - 1                                                     MIRROR     132
    XIXY = CUR(JJ2M1)**2 + CUR(JJ2)**2                                  MIRROR     133
    DELL=DELF2*(XICL-XIXY)/WNOWSQ                                       MIRROR     134
    IF(PHIAST .EQ. 0.0) GO TO 400                                       C10ASTG     14
    PHASE = AKY*(( X(J)**2 /RMSAG) + (YSQ/RMTAN)) -AKY*DELL             C10ASTG     15
    GO TO 405                                                           C10ASTG     16
400 PHASE = AKYR*(X(J)**2 + YSQ ) - AKY*DELL                            C10ASTG     17
405 CONTINUE                                                            C10ASTG     18
    SINP = SIN(PHASE)                                                   MIRROR     136
    COSP = COS(PHASE)                                                   MIRROR     137
    CURS = CUR(JJ2M1)                                                   MIRROR     138
    CUR(JJ2M1) = ( CURS*COSP - CUR(JJ2)*SINP ) * FMF                    MIRROR     139
  1 CUR(JJ2)   = ( CURS*SINP + CUR(JJ2)*COSP ) * FMF                    MIRROR     140
    IF (PHIAST.NE.0.0)WRITE(6,420)RMSAG,RMTAN                           C10ASTG     19
420 FORMAT(/,*--ASTIGMATIC PHASE ABBERATION APPLIED WITH--*,/,          C10ASTG     20
   X20X,*--SAGGITAL MIRROR RADIUS= *,E15.7,*CM*,/,                      C10ASTG     21
   X20X,*--TANGENTIAL MIRROR RADIUS= *,E15.7,*CM*,/)                    C10ASTG     22
    RETURN                                                              MIRROR     141
  2 JJ = 0                                                              MIRROR     142
    DO 3 I=1,NPY                                                        MIRROR     143
    DO 3 J=1,NPTS                                                       MIRROR     144
    JJ = JJ + 1                                                         MIRROR     145
    JJ2 = JJ * 2                                                        MIRROR     146
    JJ2M1 = JJ2 - 1                                                     MIRROR     147
    XIXY = CUR(JJ2M1)**2 + CUR(JJ2)**2                                  MIRROR     148
    DELL=DELF2*(XICL-XIXY)                                              MIRROR     149
    PHASE = AKY  * ( -DELL )                                            MIRROR     150
    SINP = SIN(PHASE)                                                   MIRROR     151
    COSP = COS(PHASE)                                                   MIRROR     152
    CURS = CUR(JJ2M1)                                                   MIRROR     153
    CUR(JJ2M1) = ( CURS*COSP - CUR(JJ2)*SINP ) * FMF                    MIRROR     154
  3 CUR(JJ2)   = ( CURS*SINP + CUR(JJ2)*COSP ) * FMF                    MIRROR     155
    RETURN                                                              MIRROR     156
 10 IF (ABS(RADCUR).LT..5) RETURN                                       MIRROR     157
    JJ = 0                                                              MIRROR     158
    DO 11 I=1,NPY                                                       MIRROR     159
    YSQ = X(I)**2                                                       MIRROR     160
    DO 11 J=1,NPTS                                                      MIRROR     161
    JJ = JJ + 1                                                         MIRROR     162
    IF(PHIAST.EQ.0.0)GO TO 480                                          C10ASTG     23
```

```
      PHASE= AKY*((X(J)**2/RMSAG) + (YSQ/RMTAN))                        CIOASTG   24
      GO TO 485                                                         CIOASTG   25
  480 PHASE = AKYR* (X(J)**2 + YSQ)                                     CIOASTG   26
  485 CONTINUE                                                          CIOASTG   27
      JJ2 = JJ * 2                                                      MIRROR   164
      JJ2M1 = JJ2 - 1                                                   MIRROR   165
      SINP = SIN(PHASE)                                                 MIRROR   166
      COSP = COS(PHASE)                                                 MIRROR   167
      CURS = CUR(JJ2M1)                                                 MIRROR   168
      CUR(JJ2M1) =   CURS*COSP - CUR(JJ2)*SINP                          MIRROR   169
   11 CUR(JJ2)    =   CURS*SINP + CUR(JJ2)*COSP                         MIRROR   170
      IF(PHIAST.NE.0.0)WRITE(6,420)RMSAG,RMTAN                          CIOASTG   28
      RETURN                                                            MIRROR   171
C ***************************************************************       MIRROR   172
C ***** TORIC MIRROR PHASE CALCULATIONS  J FORGHAM 10-4-75 *******      MIRROR   173
C ***************************************************************       MIRROR   174
  100 DO 106 I=1,NPTS                                                   MIRROR   175
      IF (X(I).GE.RANULS) GO TO 106                                     MIRROR   176
      PHI =  AKY * ((RANULS - X(I))**2) / RAUC                          MIRROR   177
      SINP = SIN(PHI)                                                   MIRROR   178
      COSP = COS(PHI)                                                   MIRROR   179
      DO 105 J=1,NPTS                                                   MIRROR   180
      K = (J-1)*NPTS                                                    MIRROR   181
      IJ = I + K                                                        MIRROR   182
      IJ2 = IJ * 2                                                      MIRROR   183
      IJ2M1 = IJ2 - 1                                                   MIRROR   184
      CURS = CUR(IJ2M1)                                                 MIRROR   185
      CUR(IJ2M1) =   CURS*COSP - CUR(IJ2)*SINP                          MIRROR   186
  105 CUR(IJ2)    =   CURS*SINP + CUR(IJ2)*COSP                         MIRROR   187
  106 CONTINUE                                                          MIRROR   188
      RETURN                                                            MIRROR   189
C   EDGE DIFFRACTION IMAGING                                            EDI        2
  300 WN = 2. * 3.1415926 / WL                                          EDI        3
      R1 = RIAIN                                                        EDI        4
      RR1 = R1 ** 2                                                     EDI        5
      R2 = RIAOUT                                                       EDI        6
      RR2 = R2 **2                                                      EDI        7
      DO 310 I=1,NPTS                                                   EDI        8
      XX = X(I)**2                                                      EDI        9
      DO 310 J=1,NPY                                                    EDI       10
      YY=X(J)**2                                                        EDI       11
      K = (J-1)*NPTS + I                                                EDI       12
      RR = XX + YY                                                      EDI       13
      IF (RR.LE.RR1.OR.RR.GE.RR2) GO TO 310                             EDI       14
      DLTPHI = WN * (RR-RR1)/RAUC                                       EDI       15
      K2 = K * 2                                                        EDI       16
      K2M1 = K2 - 1                                                     EDI       17
      SINP=SIN(DLTPHI)                                                  EDI       18
      COSP=COS(DLTPHI)                                                  EDI       19
      CURS = CUR(K2M1)                                                  EDI       20
      CUR(K2M1) = CURS*COSP - CUR(K2)*SINP                              EDI       21
      CUR(K2)    = CURS*SINP + CUR(K2)*COSP                             EDI       22
  310 CONTINUE                                                          EDI       23
      WRITE (6,320) RIAIN,RIAOUT,RAUC                                   EDI       24
  320 FORMAT (10X,33H---EDIEDIEDIEDIEDIEDIEDIEDIEDI--- /                EDI       25
     X 20X,49HAPODIZATION APPLIED WITH THE FOLLOWING PARAMETERS  ,      EDI       26
     X/20X,15HINSIDE RADIUS = ,E15.7/20X,16HOUTSIDE RADIUS = ,          EDI       27
     X E15.7/20X,21HRADIUS OF CURVATURE = , E15.7  )                    EDI       28
      RETURN                                                            EDI       29
C                                                                       EDI       30
C *****************************************************************     MIRROR   190
C ****** MIRROR €DIMPLE€ PHASE CALCULATIONS J FORGHAM 10-9-75 *******   MIRROR   191
C *****************************************************************     MIRROR   192
  200 RRTT = RIAOUT **2                                                 MIRROR   193
      DO 205 I=1,NPTS                                                   MIRROR   194
      XX = X(I)**2                                                      MIRROR   195
      DO 205 J=1,NPTS                                                   MIRROR   196
      YY = X(J)**2                                                      MIRROR   197
      RR = XX + YY                                                      MIRROR   198
      IF (RR.GT.RRTT) GO TO 205                                         MIRROR   199
      PHI =AKY * (RR-RRTT) /RAUC                                        MIRROR   200
      K = (J-1)*NPTS                                                    MIRROR   201
```

```
      IJ2 = (I + K) * 2
      IJ2M1 = IJ2 - 1
      SINP = SIN(PHI)
      COSP = COS(PHI)
      CORS = COR(IJ2M1)
      COR(IJ2M1) = CORS*COSP - COR(IJ2)*SINP
      COR(IJ2)   = CORS*SINP + COR(IJ2)*COSP
  205 CONTINUE
      RRT = -RTAOUT
      WRITE (6,600) RADC,RRT
  600 FORMAT (/,26H THE RADIUS OF CURVATURE, E11.4,40H HAS BEEN APPLIED
     AONLY OVER A RADIUS OF ,F6.3/)
      RETURN
      END
```

18. SUBROUTINE MIX

a. Purpose -- MIX calculates relaxation and pumping rates for use by subroutine KINET. The time constants which describe the various collisional processes are generated from quadratic fits to published data over a finite temperature range. The relaxation rates are then calculated from the time constants' and the cavity gas mixture ratio. This routine does not require an argument list.

Relevant Variables

TC2C	time constant for CO2 (OVO) + CO2 → CO2 + CO2
TC2N	time constant for CO2 (OVO) + CO2 → CO2 + N2
TC2O	time constant for CO2 (OVO) + O2 → CO2 + O2
TC2W	time constant for CO2 (OVO) + H2O → CO2 + H2O
TC3C	time constant for CO2 (OVO) + CO2 → CO2 (OVO) + CO2
TC3N	time constant for CO2 (OOV) + N2 → CO2 (OVO) + N2
TC3W	time constant for CO2 (OOV) + H2O → CO2 (OVO) + H2O
TPMP	time constant for N2 (V=1) + CO2 → N2 + CO2(001)
TTRD	$T_s^{-1/3}$
TTRD2	$T_s^{-2/3}$
RC2	relaxation rate for CO2 (OVO) → CO2 (000)
RC3	relaxation rate for CO2 (OOV) → CO2 (OVO)
RN2	nitrogen mispump rate (pumps CO2 bending mode)
RPUMP	pumping rate for upper level excitation

b. Relevant formalism -- The CO_2 V-V and V-T relaxation rates, the pumping rate and the nitrogen mispump rate are computed by

$$R = P_s \sum_i x_i/\tau_i \tag{143}$$

where P_S is the static pressure,

X_i are the appropriate species mole fractions,

and τ_i are their associated time constants.

The time constants, τ_i, associated with the various collisional processes are computed by an exponential quadratic fit to published data. The general form is:

$$\tau_i = \exp\left(a_i T_s^{-\frac{2}{3}} + b_i T_s^{-\frac{1}{3}} + c_i\right) \tag{144}$$

Commons Modified

/RATE/

Variables modified:

RC2	at	MIX. 28
RC3	at	MIX. 29
RN2	at	MIX. 30
RPUMP	at	MIX. 31

Commons Modified:

/MELT/

Arrays Modified:

CU incoming & outgoing field. Field is modified.

CFIL field to which CU is made orthogonal

Figure 42 is the subroutine MIX flow chart.

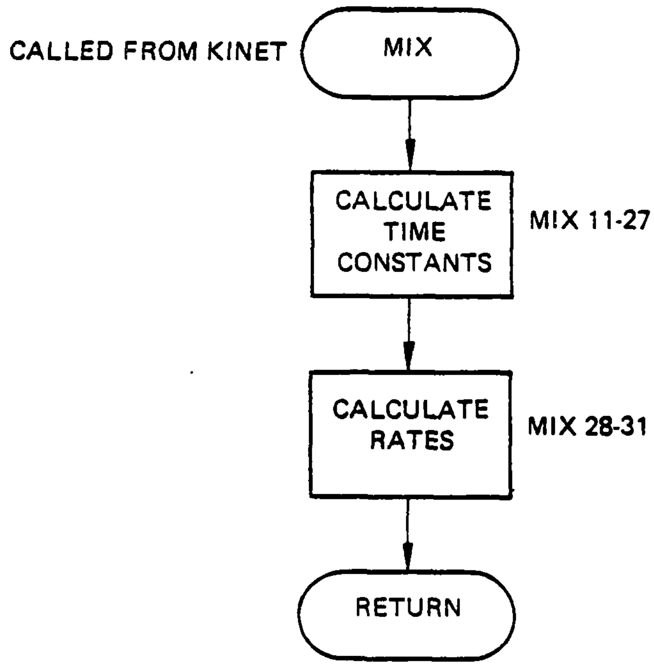


Figure 42. Subroutine MIX flow chart.

The subroutine MIX computer printout follows.

SUBROUTINE MIX   76/176   OPT=1   FIN 4.6+452   04/27/79 12.23.47

```
      SUBROUTINE MIX                                                    MIX      2
C     THIS ROUTINE CALCULATES THE CO2 V-V AND V-T RELAXATION FOR USE    MIX      3
C     IN SUBROUTINE KINET                                               MIX      4
      COMMON/PROPT/TS,PS,V,RHO,RHON,CP,GAMMA,R,B,XLAMB,HNU,CPRM         MIX      5
      COMMON/MOLES/XN2,XCO2,XH2O,XCO,XO2                                MIX      6
      COMMON/RATE/RN2,RC3,RC2,RPUMP,     RSTIM                          MIX      7
      TTHO = TS**(-.333)                                                MIX      8
      TTHO2 = TTHO**2                                                   MIX      9
C     CO2(00V)+N2= CO2(0V0)+N2                                          MIX     10
      TC3N = EXP(-393.12*TTHO2+147.69*TTHO-10.720)                      MIX     11
C     CO2(00V)+O2 = CO2(0V0)+O2                                         MIX     12
C     TC3O = TC3N                                                       MIX     13
C     CO2(00V)+CO2 = CO2(0V0)+CO2                                       MIX     14
      TC3C = EXP(-553.95*TTHO2+200.39*TTHO-15.891)                      MIX     15
C     CO2(00V)+H2O = CO2(0V0)+H2O                                       MIX     16
      TC3W = EXP(-15.895*TTHO2+.35139*TTHO-2.7323)                      MIX     17
C     CO2(0V0)+N2 = CO2+N2                                              MIX     18
      TC2N = EXP(-294.51*TTHO2+119.88*TTHO-8.6658)                      MIX     19
C     CO2(0V0)+CO2 = CO2+CO2                                            MIX     20
      TC2C = EXP(-295.96*TTHO2+120.32*TTHO-9.3265)                      MIX     21
C     CO2(0V0)+H2O = CO2+H2O                                            MIX     22
      TC2W = EXP(319.24*TTHO2-132.04*TTHO+6.9092)                       MIX     23
C     CO2(0V0)+O2 = CO2+O2                                              MIX     24
      TC2O = EXP(-195.24*TTHO2+88.380*TTHO-6.8648)                      MIX     25
```

```
C      N2(V=1)+CO2 = N2+CO2(001)
       TPMP = EXP(305.25*TTHO2-108.90*TTHO+7.0877)
       RC2= PS*(XN2/TC2N+XCO2/TC2C+XH2O/TC2W+XO2/TC2O)*1.E6
       RC3= PS*(XN2/TC3N+XCO2/TC3C+XH2O/TC3W+XO2/TC3N)*1.E6
       RN2 = PS*XCO2/TC3N*1.E6
       RPUMP = PS*(XN2+XCO2)/TPMP*1.E6
       RETURN
       END
```

19. SUBROUTINE MODER:

a. Purpose -- Subroutine MODER is designed to orthogonalize one complex field with respect to another, and to excite a higher order mode for bare resonator mode studies. The fundamental relationships are from the Siegman-Miller paper (Ref. 13). Figures 43, 44, and 45 are flow charts for the Subroutine MODER Organization.

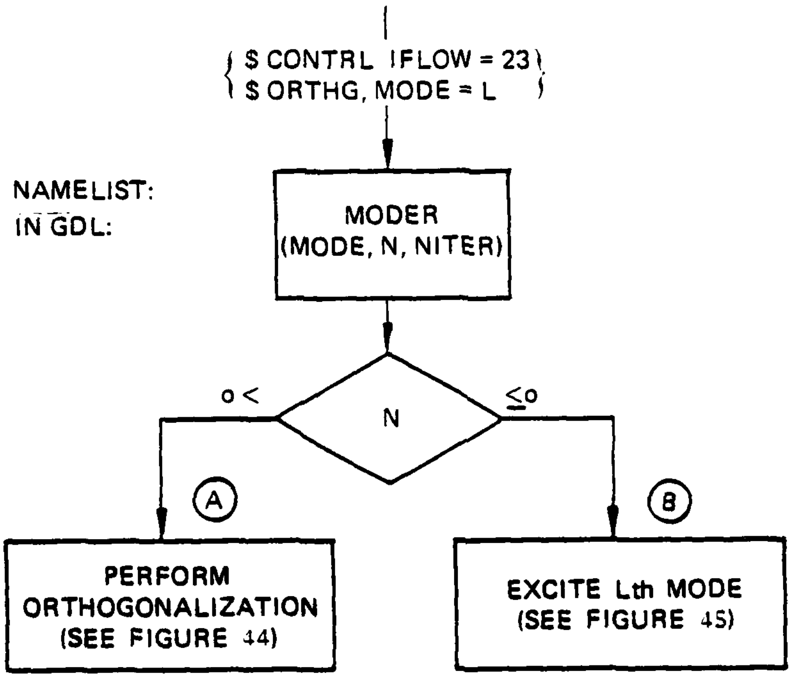


Figure 43. Subroutine MODER organization.

13. Siegman, A. E. and H. Y. Miller, "Unstable Optical Resonator Loss Calculations Using the Prony Method," Applied Optics, 9, p. 2729, 1970.

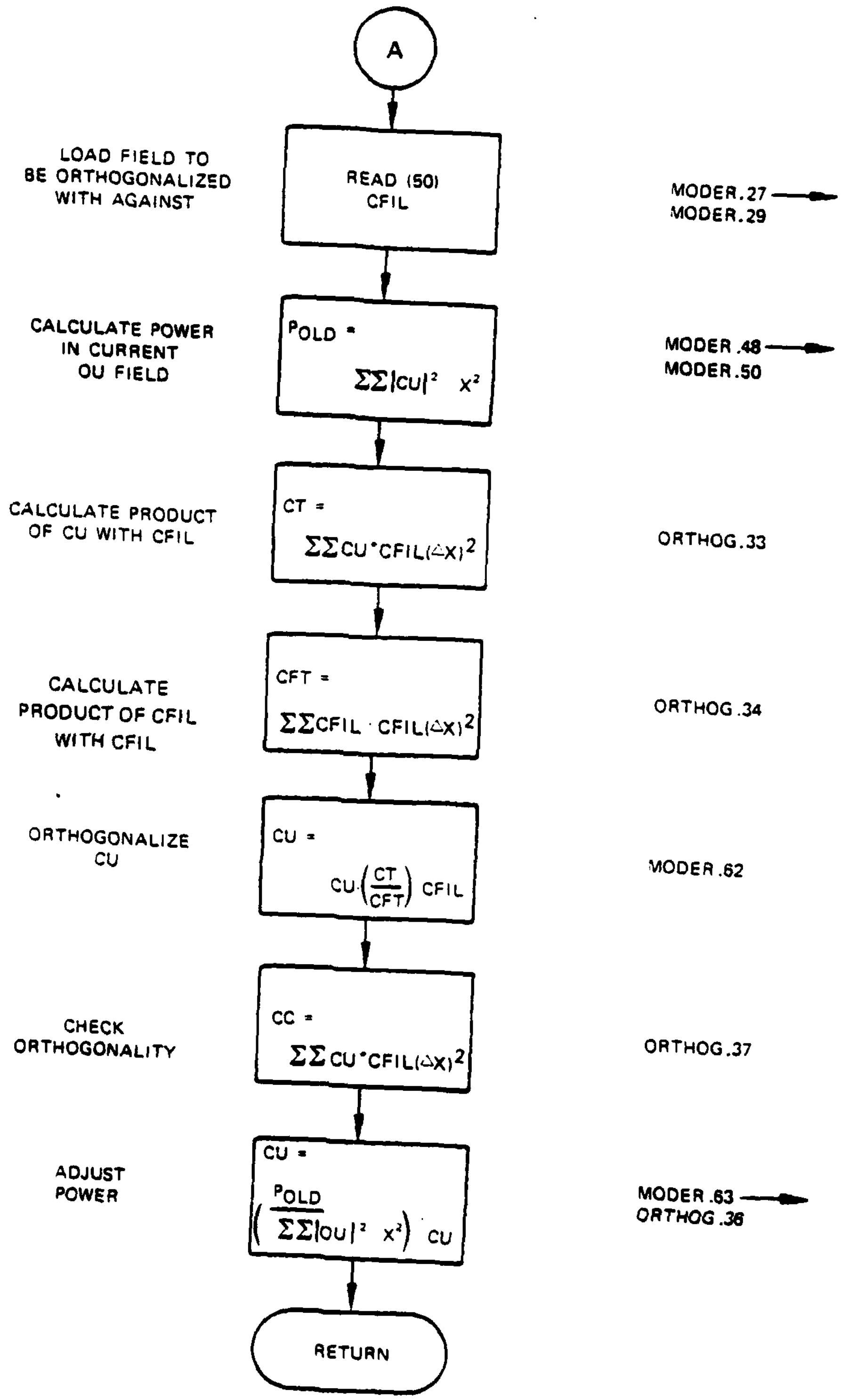


Figure 44. Perform orthogonalization.

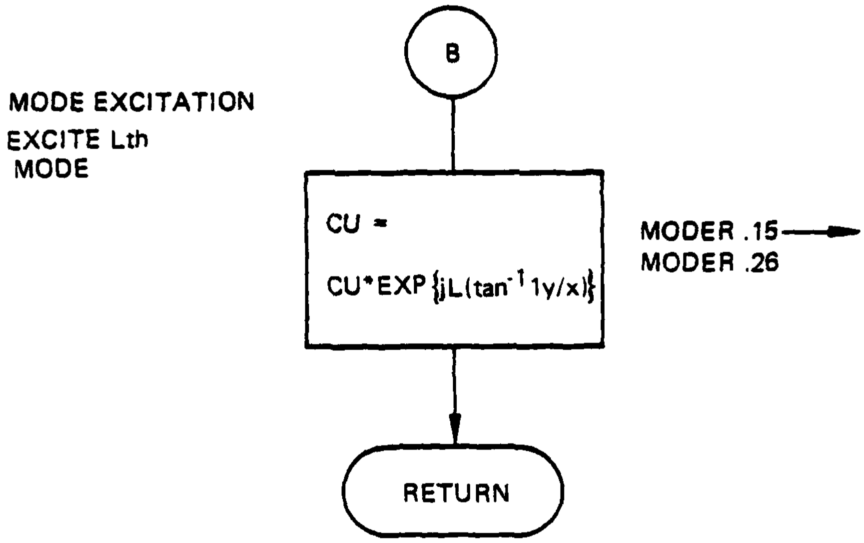


Figure 45. Mode excitation.

b. Relevant formalism -- The orthogonality condition satisifed for symmetric kernel calculations is

$$\iint_R f(x,y)g(x,y)\ dxdy = 0 \tag{145}$$

where

R = calculation region of interest

f,g = two arbitrary complex fields, described here at equispaced discrete points.

The procedure is implemented by a Gramm-Schmidt orthogonalization, to create a new field, h(x,y) from two known fields. Assume

$$h(x,y) = f(x,y) + cg(x,y) \tag{146}$$

where,

c = complex constant

g = field with which orthogonalization takes place.

then
$$\iint_R dA\, g h = 0 \tag{147}$$

So
$$C = -\left(\frac{\iint_R fg\,dA}{\iint_R gg\,dA}\right),$$

So
$$h = f - \left\{\frac{\iint_R fg\,dA}{\iint_R gg\,dA}\right\} g \qquad \forall (x,y)\,\varepsilon R$$

Numerically this becomes,

$$h_{ij} = f_{ij} - \left\{\frac{\sum_i \sum_j f_{ij} g_{ij}}{\sum_i \sum_j g_{ij}^2}\right\} g_{ij} \quad \forall (x_{ij},\, y_{ij})\ \varepsilon R \tag{148}$$

Additionally, impose the condition that

$$h_{ij} = \left[\frac{\iint |f|^2\, dA}{\iint |h|^2\, dA}\right]^{1/2} h_{ij} \tag{149}$$

then h_{ij} is the new field which is orthogonal with respect to g_{ij}, and has the same power as the initial field f.

Additionally, MODER is structured to excite the azimuthally-varying phase factor for the generation of higher order modes. In cylindrical coordinates, the modes of a bare resonator may be written as:

$$U_{ne}\ (r,\theta) = \phi_{ne}\ (r/a)\, \varepsilon^{-jl\theta} \tag{150}$$

where,

$o \leq \theta \leq 2\pi$

an arbitrary (convex mirror) scaling factor

$l = \pm 1, \pm 2, \ldots$

$n = 0, 1, 2, \ldots\ldots$

Higher order modes in bare resonators are initially excited as

$$f'(x,y) = \left[\epsilon^{-jl \tan^{-1} (y/x)}\right] f(x,y) \quad ; \quad \frac{x^2 + y^2}{a^2} \leq 1 \qquad (151)$$

and in discrete form as

$$f_{ij} = \exp \left[-jl \tan^{-1} (y_i/x_i)\right] f_{ij}$$

where f_{ij} is the SOQ complex field distribution.

c. Fortran

Argument List:

N Integer variable denoting the calculation path within the subroutine

N<0 excite the L^{th} mode and return

N>0 Perform Orthogonalization

L Order of Mode to be excited

L = 1,2, ...

Computer printouts of the MODER subroutine follow.

SUBROUTINE MODER 76/176 OPT=1 FIN 4.6+452 04/27/79 12.23.47

```
      SUBROUTINE MODER(L,R,N)
C     MODE DISCRIMINATION ROUTINE
C     THIS ROUTINE EXITES THE L-TH MODE IF N (ITERATION NUMBER) IS 0
C     AND SUPPRESSES LOWER AZIMUTHAL MODES IN SUCCESSIVE ITERATIONS
C
C  *****  THIS COPY DESIGNED TO SUPPRESS L = 0 ONLY   RDG 11-20-75 ****
C  *****  THIS COPY DESIGNED TO EXCITE L+1ST MODE    RDG 11-17-75 ***
C
      LEVEL 2, CU
      COMMON/MELT/CU(16384),CFIL(16512),X(128),WL,NPTS,NPY,DRX,DRY
      COMMON/WAY/WNOW,NREG,RAPTH
      COMPLEX CU,CFIL,CT,CFT,CC
      PI=3.141592654
      IF(N.GT.0) GO TO 100
      LP=L+1
      DO 10 I=1,NPTS
      XX=X(I)
      DO 10 J=1,NPY
      IXJ=(J-1)*NPTS+I
      YY=X(J)
      T=SPTAN(XX,YY) * LP
```

```
   10 CU(IXJ)=CU(IXJ)*CEXP(CMPLX(0.0,T))                            MODER
      WRITE(6,600) LP                                                MODER
  600 FORMAT(/,10H ***  L = ,I1,26H MODE HAS BEEN EXCITED ***,/)     MODER
      RETURN                                                         MODER
  100 CONTINUE                                                       MODER
      READ(50)  (CFIL(I),I=1,8192)                                   MODER
      REWIND 50                                                      MODER
      DO 80 I=1,NPTS                                                 MODER
      DO 80 J=1,64                                                   MODER
      IXJ=(J-1)*NPTS+I                                               MODER
      IXJ2=(128-J)*NPTS+I                                            MODER
   80 CFIL(IXJ2)=CFIL(IXJ)                                           MODER
      NOB=NPTS*NPY                                                   MODER
      P=0.0                                                          MODER
      P2=0.0                                                         MODER
      CT=CMPLX(0.0,0.0)                                              MODER
      CFT=CMPLX(0.0,0.0)                                             MODER
      DX=ABS(X(1)-X(2))                                              MODER
      DDD=(DX*WNOW)**2*(NPTS/NPY)                                    MODER
C     WRITE(6,666)DDD,R,CC,CT,AA,P,P2,IFLAG,IFLAG2                   MODER
      DO 20 I=1,NPTS                                                 MODER
      DO 20 J=1,NPY                                                  MODER
      IXJ=(J-1)*NPTS+I                                               MODER
      CT=CT+CONJG(CU(IXJ))*CFIL(IXJ)                                 MODER
      CFT=CFT+CONJG(CFIL(IXJ))*CFIL(IXJ)                             MODER
   20 P=P+CU(IXJ)*CONJG(CU(IXJ))                                     MODER
      P=P*DDD                                                        MODER
      CC=CT*DDD                                                      MODER
      WRITE(6,604) CC                                                MODER
  604 FORMAT(/,14H ***  CC     =,2G15.5,6H   ***,/)                  MODER
C     WRITE(6,666)DDD,R,CC,CT,AA,P,P2,IFLAG,IFLAG2                   MODER
      CT=CT/CFT                                                      MODER
      CC=CT                                                          MODER
      WRITE(6,604) CC                                                MODER
      CC=CFT*DDD                                                     MODER
      WRITE(6,604) CC                                                MODER
      DO 30 I=1,NPTS                                                 MODER
      DO 30 J=1,NPY                                                  MODER
      IXJ=(J-1)*NPTS+I                                               MODER
      CU(IXJ)=CU(IXJ)-CT*CFIL(IXJ)                                   MODER
   29 P2=P2+CU(IXJ)*CONJG(CU(IXJ))                                   MODER
   30 CONTINUE                                                       MODER
      P2=P2*DDD                                                      MODER
C     WRITE(6,666)DDD,R,CC,CT,AA,P,P2,IFLAG,IFLAG2                   MODER
C     WRITE(6,606) P,P2,R                                            MODER
C 606 FORMAT(/,14H ***  P,P2,R =,3G15.5,6H   ***,/)                  MODER
      SPP=SQRT(P/P2)                                                 MODER
      AA=1.0                                                         MODER
C     AA=SQRT(P/(P-CABS(CT)**2*PI*R*R))                              MODER
      WRITE(6,607) AA,SPP                                            MODER
  607 FORMAT(/,14H ***  AA,SPP =,2G15.5,6H   ***,/)                  MODER
      DO 40 I=1,NOB                                                  MODER
   40 CU(I)=CU(I)*SPP                                                MODER
C     WRITE(6,666)DDD,R,CC,CT,AA,P,P2,IFLAG,IFLAG2                   MODER
C 666 FORMAT(9H  DDD,R =,2G15.5,/,9H   CC,CT=,4G15.5,/,23H   P,P2,IFLAG, MODER
C    XIFLAG2 = ,3G15.5,2I10)                                         MODER
      RETURN                                                         MODER
      END                                                            MODER
```

20. SUBROUTINE OUTPUT

a. Purpose -- This routine generates three intensity amplitude and phase printer slice plots through the field. They are along the x-axis, the y-axis, and the "diagonal," defined by the diagram in Figure 46. Figure 47 shows the flow chart for this subroutine.

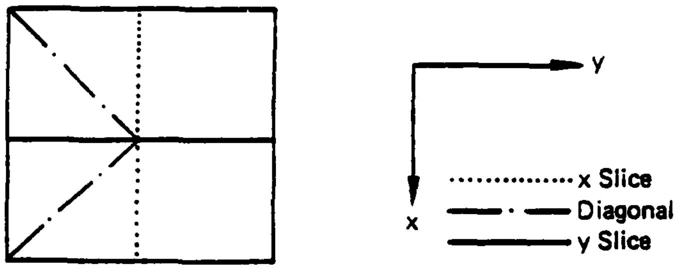


Figure 46. Intensity amplitude and phase printer slice plots.

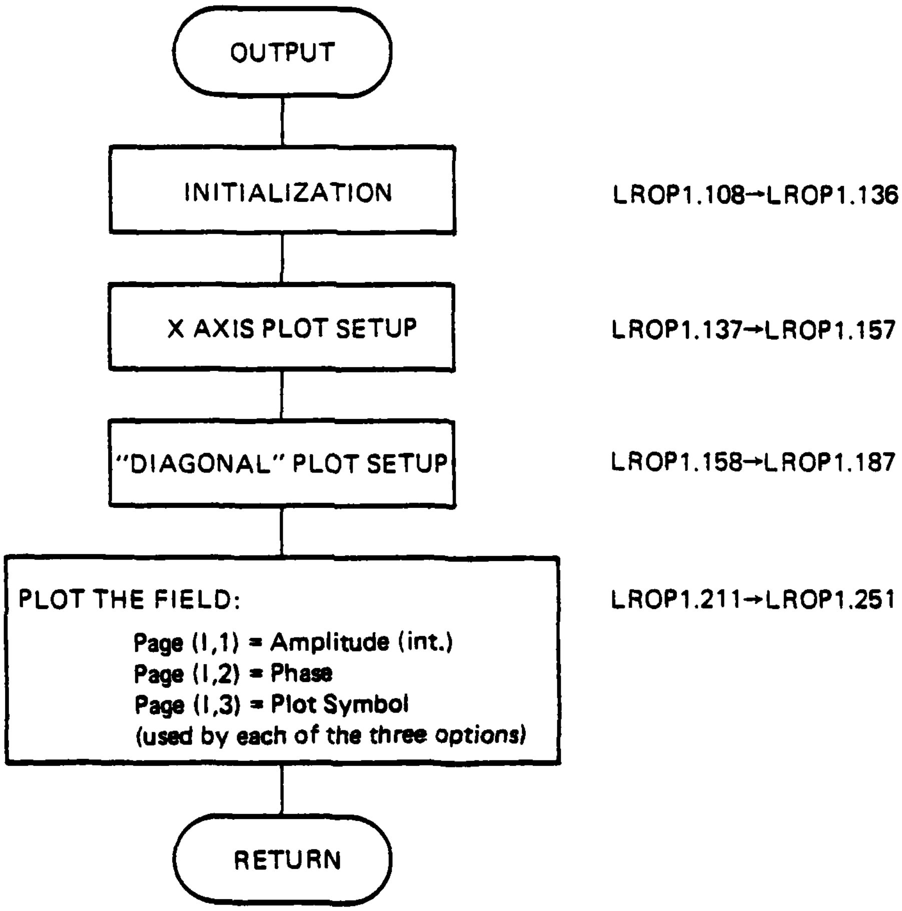


Figure 47. Subroutine OUTPUT flow chart.

b. Relevant formalism -- The slice plot uses 100 available spaces per line for plot information. The point printed shows the percent of maximum amplitude or intensity e.g., if the intensity or amplitude is 35 percent of the maximum, a symbol is printed in the 35th column. Similarly the phase is plotted from -180 to 180 degrees with zero-phase at the center. The corresponding maximum intensity amplitude is also printed out with the appropriate spatial coordinates.

c. Fortran

Argument List

CU field to be plotted

NP2 number of points in the y-direction

NP1 number of points in the x-direction

x coordinate array

N number of plots (1 to 3)

(N) = 1 → x only

2 → x and diagonal

3 → x, diagonal, and y

if N < 0, the constant J orders used is NP 2/2 instead of NP1/2. This parameter is used when gain/phase slice plots are made.

UMAX - maximum intensity amplitude of the field. It is used to establish the field point to be plotted at 100 percent.

X-AXIS - if true, the x axis plot is generated

DIAG - if true, the "diagonal" plot is generated

Y-AXIS - if true the y axis plot is generated.

No common variables are modified.

No other subroutines are called from this one.

Computer printouts for the OUTPUT subroutine follow.

```
      SUBROUTINE OUTPUR(CU,NP2,NP1,X,UMAX,DEG1,DEG2,DEG3)
C         NP1=NPTS,   NP2=NPY
C     THIS ROUTINE CONSTRUCTS PRINTER PLOTS OF RADIAL PROFILES
C     AT THREE EQUALLY SPACED ANGLES AROUND THE BEAM
C
      LEVEL 2, CU,NP2,NP1,X
      COMMON /WAY/ WNOW,NREG,RAPTR
      DIMENSION PAGE(190,3),CU(1),X(1)
      COMPLEX CU
      LOGICAL DEG1,DEG2,DEG3
C PUT IN PLOTTING SYMBOLS
      DATA POINT/1H*/,DOT/1HI/,BLANK/1H /,APOINT/1H+/
      XNP2  = NP2
      TOTAL = 360. * XNP2 / FLOAT(NP1)
      INDX2 = NP2/3
      INDX3 = (2 * NP2) / 3
      THET1 = TOTAL / 2. / XNP2
      THET2 = THET1 + FLOAT(INDX2) * TOTAL / XNP2
      THET3 = THET1 + FLOAT(INDX3) * TOTAL / XNP2
  100 NP= NP1/2
      DO 1000 K=1,3
      GO TO (10,20,30),K
   10 IF (.NOT.DEG1) GO TO 1000
      INDEX = 0
      THETA = THET1
      GO TO 1
   20 IF (.NOT.DEG2) GO TO 1000
      INDEX = INDX2 * NP1
      THETA = THET2
      GO TO 1
   30 IF (.NOT.DEG3) GO TO 1000
      INDEX = INDX3 * NP1
      THETA = THET3
    1 DO 410 I = 1,NP1
      IREF = I+INDEX
      PAGE(I,1) = CABS(CU(IREF))
      DUM1 = AIMAG(CU(IREF))
      DUM2 = REAL(CU(IREF))
      IF (DUM1.EQ.0.0.AND.DUM2.EQ.0.0) GO TO 412
  411 PAGE(I,2) = 57.3*ATAN2(DUM1,DUM2)
      GO TO 410
  412 PAGE(I,2) = 0.0
  410 UMAX = AMAX1(UMAX,PAGE(I,1))
      WRITE (6,520) THETA
  520 FORMAT(33H1     CU(I,J) PLOTTED RADIALLY AT ,F7.2,9H  DEGREES )
      IF (K.NE.1) GO TO 1001
      UMAXP=UMAX
      IF(NREG.NE.0.AND.N.GT.0)UMAXP=UMAX/WNOW
 1001 IF (UMAX.EQ.0.0) UMAX = 1.0
      SCALE1 = 100.0/UMAX
      SCALE2 = 50.0/180.0
C PRINT AXES
      WRITE (6,460)UMAXP
  460 FORMAT (1H ,T2,1H0,T27,2H25,T52,2H50,T58,13HMAGNITUDE (*),T77,2H75
     *,T101,5H100 =,G12.4)
      WRITE (6,450)
  450 FORMAT (1H ,T2,4H-180,T26,3H-90,T52,1H0,T58,15HPHASE ANGLE (+),T76
     *,3H+90,T101,4H+180,7X,1HR,/X,4HAMPL,4X,5HPHASE)
C USE PAGE(L,3) AS PRINTING LINE -- FIRST BLANK IT
      DO 420 L = 1,104
  420 PAGE(L,3) = BLANK
C PRINT A LINE FOR EACH VALUE OF I
      DO 430 I = 1,NP1
      DO 440 L = 1,101,25
  440 PAGE(L,3) = DOT
      PAGE(51.5+SCALE2*PAGE(I,2),3) = APOINT
      RELAMP = SCALE1 * PAGE(I,1)
      PAGE(1.5 + RELAMP  ,3) = POINT
      WRITE (6,470)  (PAGE(L,3),L=1,104),  X(I),RELAMP   ,PAGE(I,2)
  470 FORMAT (1H ,104A1,3F9.2)
```

```
      PAGE(1.5+SCALE1*PAGE(I,1),3) = BLANK
  430 PAGE(51.5+SCALE2*PAGE(I,2),3) = BLANK
 1000 CONTINUE
      RETURN
C*****************************************************************C
      END
```

SUBROUTINE OUTPUT   76/176   OPT=1   FIN 4.6+452   04/27/79 12.23.47

```
      SUBROUTINE OUTPUT(CU,NP2,NP1,X,N,UMAX,XAXIS,DIAG,YAXIS)
C           NP1=NPTS,   NP2=NPY
C     THIS ROUTINE CONSTRUCTS PRINTER SLICE PLOTS OF THE COMPLEX FIELD
C     ALONG (1) THE Y AXIS, (2) ALONG A DIAGONAL AND (3) ALONG THE
C     X-AXIS THROUGH THE FIELD.  Y-AXIS PLOTS ONLY FOR CAVITY PARAMETERS
      LEVEL 2, CU,NP2,NP1,X
      COMMON /WAY/ WNOW,NREG,RAPTR
      COMMON /PLTSIG/ PLOTSG
      COMPLEX CU
      LOGICAL XAXIS,DIAG,YAXIS
      DIMENSION PAGE(190,3),IDIAG(128),XP(190),YP(190),CU(1),X(1)
      DIMENSION IMAG(3) , IINT(3) , ITITL(3)
C PUT IN PLOTTING SYMBOLS
      DATA POINI/1H*/,DOT/1HI/,BLANK/1H /,APOINT/1H+/,POINA/1H@/
      DATA IMAG/4HMAGN,4HITUD,4HE(@)/,IINT/4HINTE,4HNSIT,4HY(*)/
      IF (PLOTSG.GT.0.) GO TO 100
      POINT = POINA
      DO 110 IP=1,3
  110 ITITL(IP) = IMAG(IP)
      GO TO 150
  100 POINT = POINI
      DO 120 IP=1,3
  120 ITITL(IP) = IINT(IP)
  150 CONTINUE
      NP= NP1/2
      IF(N.LT.0)NP=NP2/2
      NN=IABS(N)
      DO 1000 K=1,NN
      GO TO (1,2,3),K
    1 IF (.NOT.XAXIS) GO TO 1000
      NP2X=NP1*(NP-1)
C     X-AXIS PLOT (I.E. Y=0)
      DO 410 I = 1,NP1
      XP(I)=X(I)
      YP(I)=X(NP)
      IF(N.LT.0)YP(I)=0.0
      IREF = I+NP2X
      PAGE(I,1) = CABS(CU(IREF))
      IF (PLOTSG.GT.0.) PAGE(I,1)=PAGE(I,1)**2
      DUM1 = AIMAG(CU(IREF))
      DUM2 = REAL(CU(IREF))
      IF (DUM1.EQ.0.0.AND.DUM2.EQ.0.0) GO TO 412
  411 PAGE(I,2) = 57.3*ATAN2(DUM1,DUM2)
      GO TO 410
  412 PAGE(I,2) = 0.0
  410 UMAX = AMAX1(UMAX,PAGE(I,1))
      UMAXP=UMAX
      IF(NREG.NE.0.AND.N.GT.0.AND.PLOTSG.LT.0.) UMAXP=UMAX/WNOW
      IF(NREG.NE.0.AND.N.GT.0.AND.PLOTSG.GT.0.) UMAXP=UMAX/WNOW**2
      GO TO 1001
    2 IF (.NOT.DIAG) GO TO 1000
      DO 10 I=1,NP1
      I1=NP1+1-I
      IDIAG(I1)=(MIN0(I1,I)-1)*NP1+I1
   10 CONTINUE
C     DIAGONAL PLOT (I.E. X=Y)
      DO 510 I = 1,NP1
```

```
      XP(I)=X(I)
      NYP=NP1-I+1
      IYP=MIN0(I,NYP)
      YP(I)=X(IYP)
      IREF = IDIAG(I)
      PAGE(I,1) = CABS(CU(IREF))
      IF (PLOTSG.GT.0.) PAGE(I,1)=PAGE(I,1)**2
      DUM1 = AIMAG(CU(IREF))
      DUM2 = REAL(CU(IREF))
      IF (DUM1.EQ.0.0.AND.DUM2.EQ.0.0) GO TO 512
  511 PAGE(I,2) = 57.3*ATAN2(DUM1,DUM2)
      GO TO 510
  512 PAGE(I,2) = 0.0
  510 IF(PLOTSG.LT.0.) UMAX = AMAX1(UMAX,PAGE(I,1))
      IF (PLOTSG.GT.0.) GO TO 935
      WRITE(6,520)
  520 FORMAT(75H1AMPLITUDE,PHASE PLOTTED ALONG A DIAGONAL THROUGH THE CE
     XNTER OF UCALC            )
      GO TO 1001
  935 WRITE(6,634)
  634 FORMAT(75H1INTENSITY,PHASE PLOTTED ALONG A DIAGONAL THROUGH THE CE
     XNTER OF UCALC            )
      GO TO 1001
C     Y-AXIS PLOT (I.E. X=0)
    3 IF (.NOT.YAXIS) GO TO 1000
      DO 610 I = 1,NP2
      XP(I)=X(NP)
      YP(I)=X(I)
      IREF = NP+(I-1)*NP1
      PAGE(I,1) = CABS(CU(IREF))
      IF (PLOTSG.GT.0.) PAGE(I,1)=PAGE(I,1)**2
      DUM1 = AIMAG(CU(IREF))
      DUM2 = REAL(CU(IREF))
      IF (DUM1.EQ.0.0.AND.DUM2.EQ.0.0) GO TO 612
  611 PAGE(I,2) = 57.3*ATAN2(DUM1,DUM2)
      GO TO 610
  612 PAGE(I,2) = 0.0
  610 IF (PLOTSG.LT.0.) UMAX = AMAX1(UMAX,PAGE(I,1))
      IF (PLOTSG.GT.0.) GO TO 3264
      WRITE(6,620)
  620 FORMAT(70H1AMPLITUDE,PHASE PLOTTED IN Y-DIRECTION THROUGH CENTER O
     XF UCALC           )
      GO TO 1001
 3264 WRITE(6,852)
  852 FORMAT(70H1INTENSITY,PHASE PLOTTED IN Y-DIRECTION THROUGH CENTER O
     XF UCALC           )
 1001 IF (UMAX.EQ.0.0) UMAX = 1.0
      SCALE1 = 100.0/UMAX
      SCALE2 = 50.0/180.0
C PRINT AXES
      WRITE (6,460) ITITL, UMAXP
  460 FORMAT (1H ,T2,1H0,T27,2H25,T52,2H50,T58,3A4                ,T77,2H75
     *,T101,5H100 =,G12.4)
      IF(N.GT.0)WRITE (6,450)
  450 FORMAT (1H ,T2,4H-180,T26,3H-90,T52,1H0,T58,15HPHASE ANGLE (*),T76
     *,3H+90,T101,4H+180,/X,1HR,8X,1HX,8X,1HY)
      IF(N.LT.0)WRITE (6,451)
  451 FORMAT (1H ,T2,4H-180,T26,3H-90,T52,1H0,T58,15HPHASE ANGLE (*),T76
     *,3H+90,T101,4H+180,6X,4HANEP,6X,1HY)
C USE PAGE(L,3) AS PRINTING LINE -- FIRST BLANK IT
      DO 420 L = 1,130
  420 PAGE(L,3) = BLANK
C PRINT A LINE FOR EACH VALUE OF I
      NER=NP1
      IF(K.EQ.3)NER=NP2
      IF(N.LT.0)GO TO 301
      DO 430 I = 1,NER
      DO 440 L = 1,101,25
  440 PAGE(L,3) = DOT
      PAGE(51.5+SCALE2*PAGE(I,2),3) = APOINT
      PAGE(1.5+SCALE1*PAGE(I,1),3) = POINT
```

```
      R=SQRT(XP(I)**2+YP(I)**2)
      WRITE (6,470)   (PAGE(L,3),L=1,104),R,XP(I),YP(I)
  470 FORMAT (1H ,104A1,3F9.2)
      PAGE(1.5+SCALE1*PAGE(I,1),3) = BLANK
  430 PAGE(51.5+SCALE2*PAGE(I,2),3) = BLANK
      GO TO 1000
  301 DO 330 I = 1,NER
      DO 340 L = 1,101,25
  340 PAGE(L,3) = DOT
      PAGE(51.5+SCALE2*PAGE(I,2),3) = APOINT
      PAGE(1.5+SCALE1*PAGE(I,1),3) = POINT
      WRITE (6,370)   (PAGE(L,3),L=1,104),XP(I),YP(I)
  370 FORMAT (1H ,104A1,2F9.2)
      PAGE(1.5+SCALE1*PAGE(I,1),3) = BLANK
  330 PAGE(51.5+SCALE2*PAGE(I,2),3) = BLANK
 1000 CONTINUE
      RETURN
C*****************************************************************C
      END
```

21. SUBROUTINE PLTOT

Subroutine PLTOT is called at the end of subroutine QUAL to calculate and generate a printer plot of far field power versus radial distance in Rλ/D units. The integrated fractional power to several far field radii are calculated by multiple calls to subroutine POWWOW. The power and radius values are stored by PLTOT in array form. The arrays are then tabulated. A simple printer plot is also generated without the necessity of an interpolation scheme or other formal calculations.

Figure 48 is the Subroutine PLTOT flow chart and is followed by the PLTOT computer printouts.

Argument List

DB	near field beam diameter
DX	grid spacing in far field, Rλ/D units
IMAX	number of field points across grid
IPLT	flag - not used
PT	total near field power
RMAX	not used
TITLE	run identification
WL	wavelength
XCEN	X-position of center of interest
XX	X-position array
YCEN	Y-position of center of interest
Z	far field intensity array
ZMAX	not used

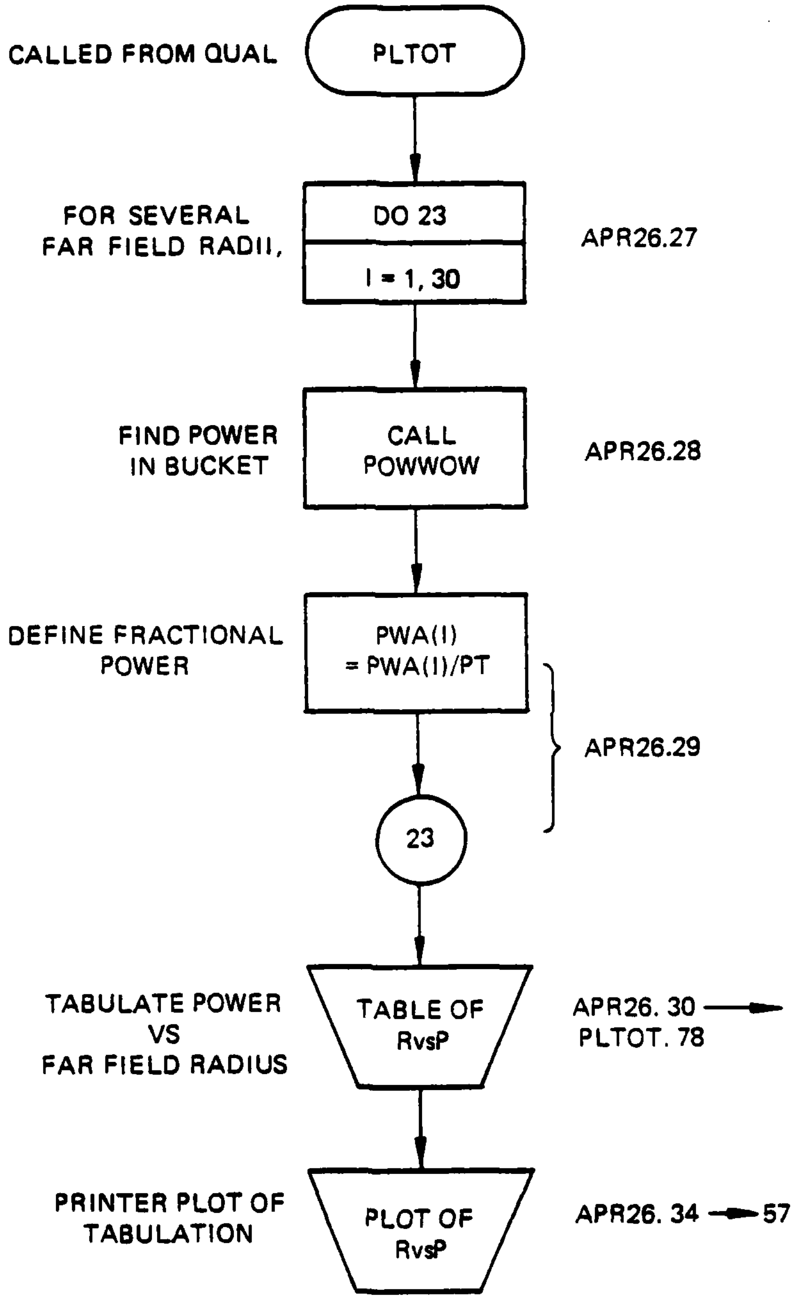


Figure 48. Subroutine PLTOT flow chart.

Relevant variables

IPAGE	Hollerith character string comprising a single vertical position of printer plots
PWA	fractional power array corresponding to RRD
RRD	radial distance array corresponding to various far field bucket sizes.

SUBROUTINE PLTOT 76/176 OPT=1 FIN 4.6+452 04/27/79 12.23.47

```
      SUBROUTINE PLTOT ( IMAX, DX, XX, ZMAX, RMAX, Z, IPLT, TITLE,       PLTOT       2
     1 PT , XCEN , YCEN , DB , WL )                                     APR26      21
      LEVEL 2, WL                                                       APR26      22
C     THIS ROUTINE (1) MAKES AN ISO-INTENSITY PLOT OF THE FAR FIELD     PLTOT       4
C     SPOT AND (2) CALCULATES AND PLOTS THE POWER VERSUS FAR FIELD      PLTOT       5
C     RADIUS.                                                           PLTOT       6
      LEVEL 2, IMAX,XX,Z                                                PLTOT       7
      DIMENSION XX(1), Z(  1  ) , TITLE(20)                             PLTOT       8
      DIMENSION PWA(30), RRD(30) ,IPAGE(101)                            APR26      23
      DATA RRD/.2,.4,.5,.6,.7,.8,.9,1.,1.1,1.2,1.3,1.4,1.5,1.6,         APR26      24
     X 1.7,1.8,1.9,2.,2.1,2.2,2.3,2.4,2.5,2.6,2.7,2.8,2.9,3.,4.,5./     APR26      25
      DATA IBLNK/4H    /,II/1HI/,IPT/1H./                               APR26      26
C     DIMENSION LAB(5)                                                  PLTOT      10
C     DATA  NOZ,LAB /  6,80,60,40,20,10,5,2,1,12*0 /                    PLTOT      11
C     CALL DATE(MNTH,DAY,YEAR)                                          PLTOT      12
C     CALL HCLOCK(HR,MIN,SEC)                                           PLTOT      13
C     GO TO (32,51),IPLT                                                PLTOT      14
C     PLOT FAR FIELD ISO-INTENSITIES                                    PLTOT      15
C     XSCL=3./RMAX                                                      PLTOT      16
C     CALL INIT(&SIZE&,8.,10.)                                          PLTOT      17
C     CALL PLOT(3.5,3.5,23)                                             PLTOT      18
C     CALL TXSIZ(.2,.13)                                                PLTOT      19
C     CALL TXPLT(0.,5.,0.,0)                                            PLTOT      20
C     WRITE(98,1)                                                       PLTOT      21
C   1 FORMAT(36H  FAR FIELD ISO-INTENSITY CONTOURS   )                  PLTOT      22
C     CALL TXSIZ(.12,.08)                                               PLTOT      23
C     CALL TXPLT(0.,4.5,0.,0)                                           PLTOT      24
C     WRITE(98,2) TITLE,RMAX,MNTH,DAY,YEAR,HR,MIN,SEC                   PLTOT      25
C   2 FORMAT(1X,20A4//29H THE LARGEST RADIUS PLOTTED =,F4.1,9HR*LAMB/D / PLTOT     26
C    1/5HDATE ,A2,1H/,A2,1H/,A2,10X,5HTIME ,A2,1H.,A2,1H.,A2)           PLTOT      27
C     CALL SYMBOL(0.,0.,.15,3,0.,-1)                                    PLTOT      28
C     DO 190 I=1,4                                                      PLTOT      29
C     ADP=.04*I                                                         PLTOT      30
C     AUP=.03*I                                                         PLTOT      31
C     CALL DASH(ADP,AUP)                                                PLTOT      32
C     RR=RMAX*I/4 *XSCL                                                 PLTOT      33
C 190 CALL CIRC(&RAD&,RR,&CEN&,0.,0.)                                   PLTOT      34
C     CALL NODASH                                                       PLTOT      35
C     CALL ISO( XX,XX,Z,ZMAX,0.,IMAX,IMAX,XCEN,YCEN,XSCL,NOZ,LAB,IMAX)  PLTOT      36
C     CALL FINI                                                         PLTOT      37
C     IF (IPLT.EQ.3) GO TO 51                                           PLTOT      38
C     PLOT POWER VS. R*LAMBDA/D. THIS IS DONE ABOUT EITHER THE CENTROID PLTOT      39
C     OR PEAK INTENSITY WHICHEVER DEMONSTRATES MAXIMUM PERFORMANCE.     PLTOT      40
C  32 CALL INIT(&SIZE&,8.,10.)                                          PLTOT      41
C     CALL PLOT(1.5,1.,23)                                              PLTOT      42
C     CALL AXIS(0.,0.,11HRADIUS-WL/D,-11,4.,0.,0.,RMAX/4.)              PLTOT      43
C     CALL AXIS(0.,0.,13HPERCENT POWER,13,5.,90.,0.,20.)                PLTOT      44
C     CALL GRID(0.,0.,4.,16,5.,20)                                      PLTOT      45
C     CALL TXSIZ(.15,.09)                                               PLTOT      46
C     CALL TXPLT(2.,8.,0.,0)                                            PLTOT      47
C     WRITE(98,50) TITLE,MNTH,DAY,YEAR,HR,MIN,SEC                       PLTOT      48
C  50 FORMAT(28H  FAR FIELD QUALITY   (FFT)  //20A4//5HDATE ,A2,1H/,A2,1H PLTOT    49
C    1/,A2,10X,5HTIME ,A2,2(1H.,A2))                                    PLTOT      50
C     CALL MOVEA(0.,0.)                                                 PLTOT      51
   51 IRAD=RMAX*2.                                                      PLTOT      52
C     PRINT POWER VS. R*LAMBDA/D                                        PLTOT      53
      WRITE(6,22) TITLE                                                 PLTOT      54
```

```
   22 FORMAT(///,1X,20A4,///,3X,5(2X,14HR,.P(FRACTION)),/)              PLTOT    55
C       DO 23 I=1,IRAD                                                  PLTOT    56
C       DO 25 J=1,5                                                     PLTOT    57
C       RRO(J)=.1*((I-1)*5.+J)                                          PLTOT    58
C       CALL POWWOW ( IMAX, DX, XX, Z, XCEN, YCEN, RRO(J), PWA(J) )     PLTOT    59
C       PWA(J) = PWA(J) / PT                                            PLTOT    60
C  25 IF (IPLT.LE.1) CALL LINEA(RRO(J)*4./RMAX PWA(J)*5.)               PLTOT    61
C  23 WRITE(6,24) (RRO(K),PWA(K),K=1,5)                                 PLTOT    62
      RROX=0.0                                                          PLTOT    63
      DRRO=0.1                                                          PLTOT    64
      DO 23 I=1,30                                                      APR26    27
      CALL POWWOW(IMAX,DX,XX,Z,XCEN,YCEN,RRO(I),PWA(I))                 APR26    28
   23 PWA(I) = PWA(I) / PT                                              APR26    29
      DO 25 I=1,6                                                       APR26    30
      J1 = (I-1)*5 + 1                                                  APR26    31
      J2 = J1 + 4                                                       APR26    32
   25 WRITE (6,24) (RRO(K),PWA(K),K=J1,J2)                              APR26    33
   24 FORMAT(5(4X,F4.1,F8.5))                                           PLTOT    78
   26 CONTINUE                                                          PLTOT    79
C       IF (IPLT.LE.1) CALL FINI                                        PLTOT    80
      WRITE(6,1100)  WL,DB                                              APR26    34
 1100 FORMAT(1H1/////46X,18HPERCENT TOTAL FLUX /45X,3HWL=,F8.6,4H  DB=  APR26    35
     X,F6.2/2X,6HRRAD  0,23X,2H25,23X,2H50,23X,2H75,22X,3H100   )       APR26    36
      DO 1310 I=2,100                                                   APR26    37
 1310 IPAGE(I) = IBLNK                                                  APR26    38
      IRAD = 1                                                          APR26    39
      DO 1320 LINE=1,51                                                 APR26    40
      IPAGE(1)=II                                                       APR26    41
      IPAGE(26)=II                                                      APR26    42
      IPAGE(51)=II                                                      APR26    43
      IPAGE(76)=II                                                      APR26    44
      IPAGE(101)=II                                                     APR26    45
      RAD = (LINE-1)*.1                                                 APR26    46
      PCR = RRO(IRAD)                                                   APR26    47
      IF (ABS(RAD-PCR).GT..01) GO TO 1315                               APR26    48
      INDEX = 1.5 + PWA(IRAD)*100.                                      APR26    49
      IRAD = IRAD + 1                                                   APR26    50
      IPAGE(INDEX) = IPT                                                APR26    51
      WRITE(6,1110) RAD ,IPAGE                                          APR26    52
 1110 FORMAT (1X,F4.2,2X,101A1   )                                      APR26    53
      IPAGE(INDEX)=IBLNK                                                APR26    54
      GO TO 1320                                                        APR26    55
 1315 WRITE (6,1110) RAD,IPAGE                                          APR26    56
 1320 CONTINUE                                                          APR26    57
      RETURN                                                            PLTOT    83
      END                                                               PLTOT    84
```

22. SUBROUTINE POWWOW

Calls: N/A

Called by: QUAL

a. Purpose -- POWWOW is called by QUAL to apply an aperture to the far field intensity field for computing power in the bucket. Figure 49 shows the POWWOW flow chart, followed by the POWWOW computer printouts.

POWWOW passes the intensity field, x and y centroid locations, and bucket size. It returns the power in the bucket in parameter PRB (PWR).

POWOW defines a radius function, RD, for converting rectangular coordinates to a radius bucket size. Each x,y coordinate is searched to determine if it is within the bucket. If so, the power at that point is added to the sum for the bucket.

After all locations have been checked, control is returned to QUAL along with the power number.

b. Relevant formalism -- Each grid point (X,Y) lies at the center of a square ΔX on a side. In the logic to determine whether a point falls within the radius of interest, an attempt is made to account for grids which fall partially within the radius, RAD. These points are weighted between 0 and 1 according to

$$P = (RAD-R_{min})/(R_{max}-R_{min})$$

where

P is the weighting factor,

R_{max} is the radius to the furthest corner of the grid, and

R_{min} is the radius to the nearest corner of the grid.

All grid points with R_{max} less than RAD are given a weight of 1, all grid points with R_{min} greater than RAD are weighted 0.

Argument List

AA	far field intensity array
DX	separation of far field points
NPTS	number of array points in one dimension
PWR	power in the bucket - returned to calling routine
RAD	radius of far field bucket
XAR	X-position array for intensity field
XCEN	X-position of center of interest
YCEN	Y-position of center of interest

Relevant Variables

DS	area associated with a grid point
PER	weighting factor for a grid point, between 0 and DS
X	X-position of grid point
Y	Y-position of grid point

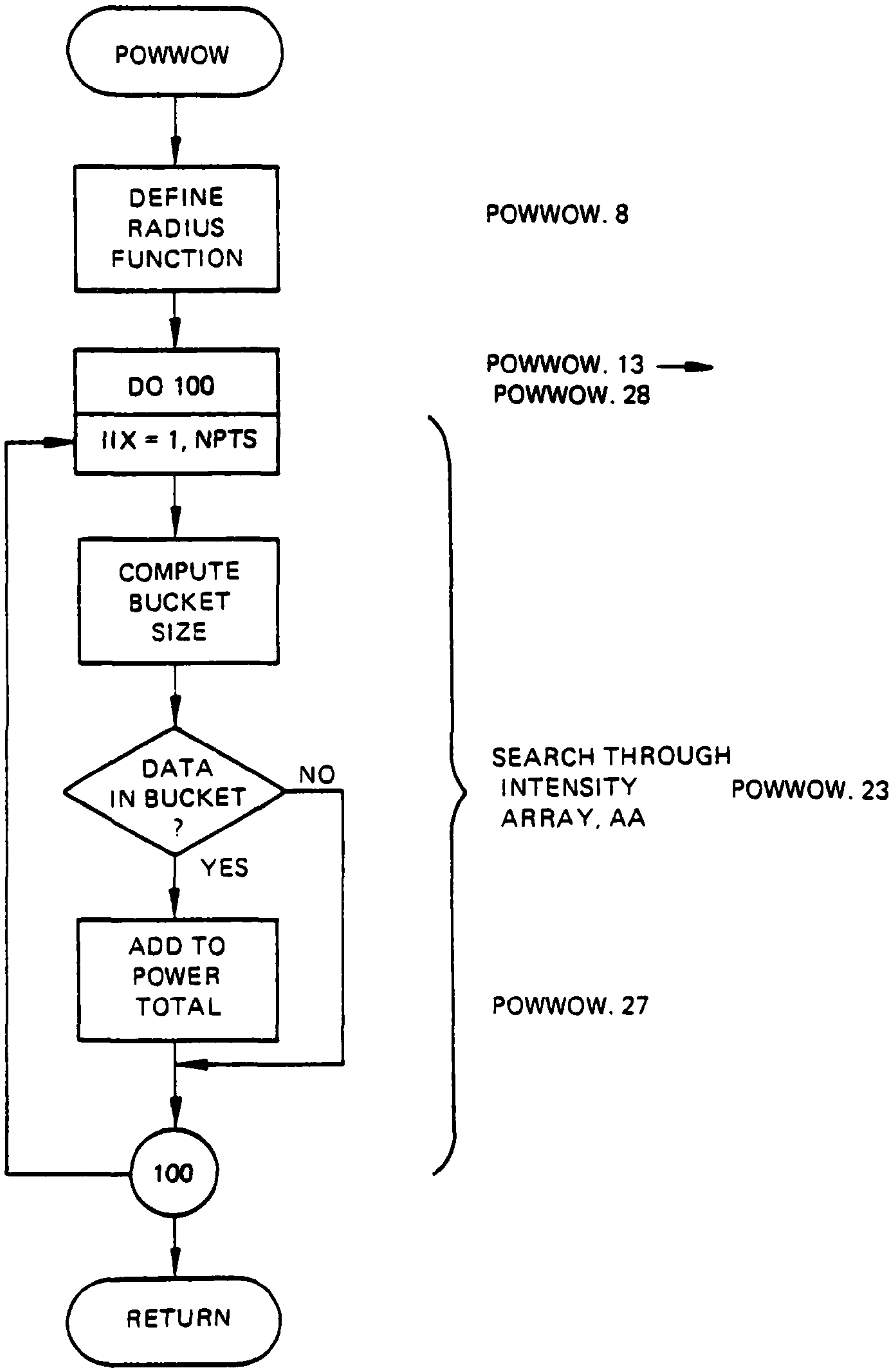


Figure 49. Subroutine POWWOW flow chart.

SUBROUTINE POWWOW 76/176 OPT=1 FIN 4.6+452 04/27/79 12.23.47

```
      SUBROUTINE POWWOW (    NPTS,      DX,        XAR,       AA,
     1   XCEN,      YCEN,      RAD,       PWR )
C     THIS ROUTINE APPLIES AN APERTURE TO THE FAR FIELD INTENSITY
C     PATTERN FOR DETERMINING POWER VS. R*LAMBDA/D
      LEVEL 2, NPTS,XAR,AA
      DIMENSION    XAR(1),    AA(  1  )
      RD(XX,YY,IX,IY)=SQRT((ABS(XX)+IX*DX/2.)**2+(ABS(YY)+DY/2.*IY)**2)
      PWR = 0.
      DY=DX
      DS = DX ** 2
      DO 100 IIX=1,NPTS
      X=XAR(IIX)-XCEN
      DO 100 IIY=1,NPTS
      Y=XAR(IIY)-YCEN
      RPP=RD(X,Y,1,1)
      RMM=RD(X,Y,-1,-1)
      RMP=RD(X,Y,-1,1)
      RPM=RD(X,Y,1,-1)
      PER=DS
      RMAX=AMAX1(RPP,RMM,RMP,RPM)
      IF(RMAX.LE.RAD) GO TO 100
      PER = 0.
      RMIN=AMIN1(RPP,RMM,RMP,RPM)
      IF(RMIN.GE.RAD) GO TO 100
      PER=(RAD-RMIN)/(RMAX-RMIN)*DS
  100 PWR=PWR+AA(IIX+(IIY-1)*NPTS)*PER
      RETURN
      END
```

23. SUBROUTINE QUAL

Called by: MAIN

Calls: TILT
STEP
CENBAR
POWWOW

QUAL, entered with a call from MAIN, is used to calculate quality of complex field. Figure 50 is the flow chart for the QUAL subroutine. Subroutine QUAL computer printouts follow Figure 50. A decision is made whether to use the COMMON complex field or whether to read one in from tape. A decision is then made as to whether or not to save whatever input complex field is used. This is for later restoration.

Variables are initialized and, based on the call statement input variables, a decision is made whether or not to apply a phase correction to the complex field, that is, should tilt and/or spherical components be removed? If not, QUAL branches to the lens section.

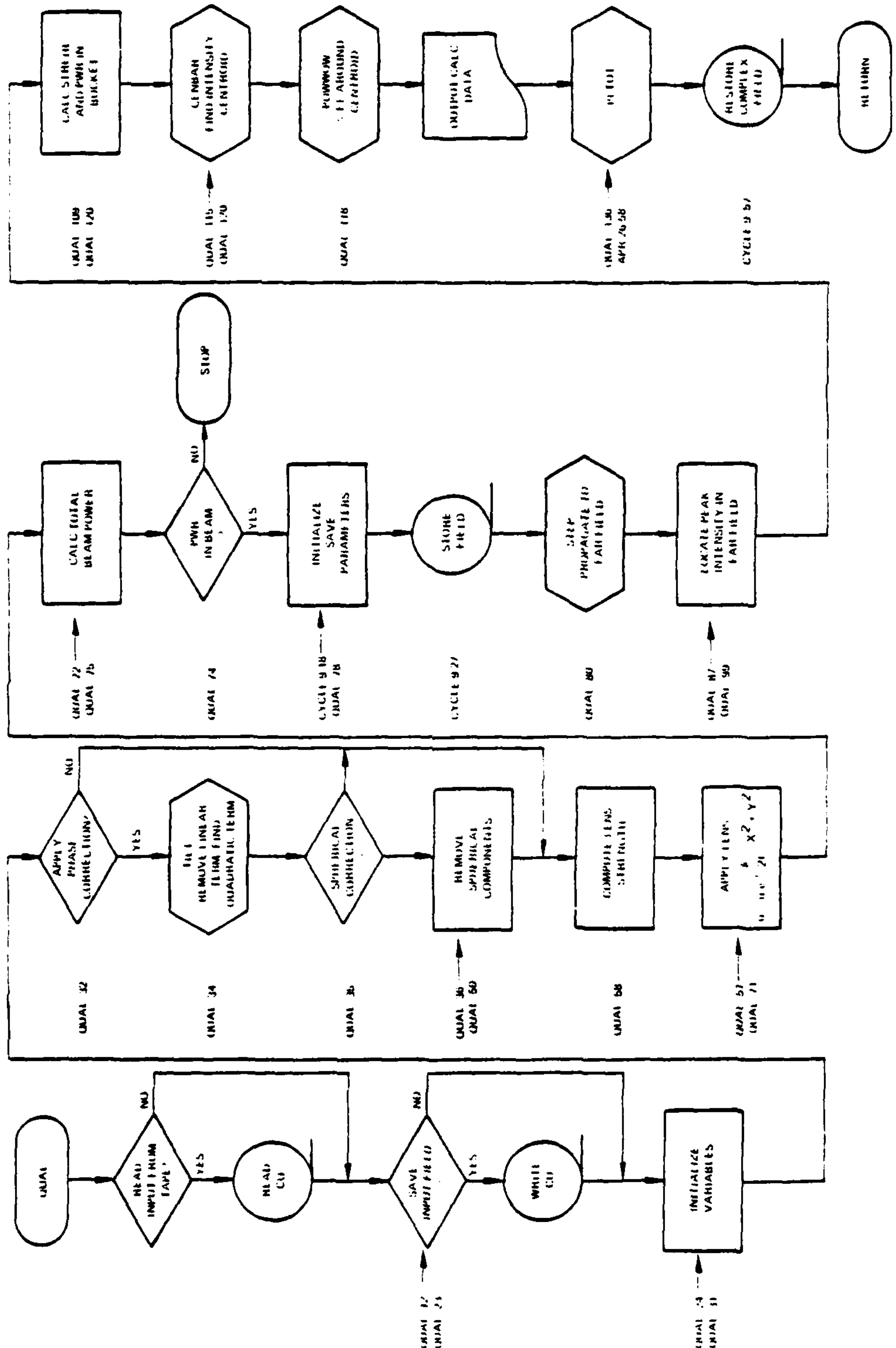


Figure 50. Subroutine QUAL flow chart.

If yes, then a call is made to subroutine TILT and the linear and quadratic phase components are removed. If spherical components are to be removed, then this is done. If not, control passes to the lens section.

The lens strength required to bring the beam down to a specified radius is computed. This is then applied to the field, CU, via the relation

$$U = U \exp \left| i \frac{k}{2f} (x^2 + y^2) \right| \tag{152}$$

The total beam power as transformed by the lens is then calculated. If there is no power in the transformed beam, an error message is output and the job stopped. Otherwise, some saving parameters are initialized and the transformed field is saved on tape.

Subroutine STEP is called to take the transformed beam to the far field. The location of the far field peak intensity is found. Strehl and power in the bucket are calculated. Subroutine CENBAR is called to find the percent of far field centroid (intensity). Subroutine POWWOW is called to find the percent of far field power in a given radius around the centroid. All of the calculated data is printed and subroutine PLTOT is called for beam quality plots.

QUAL then restores the complex field to what it was at entry and control is returned to MAIN.

SUBROUTINE QUAL   76/176   OPT=1   FIN 4.6+452   04/27/79 12.23.47

```
      SUBROUTINE QUAL ( IPHASE , ISAVE , IPLT, TITLE, RB, ANS, DB ,RF)   QUAL        2
C     FAR FIELD QUALITY ALGORITHM                                       QUAL        3
C     THIS ROUTINE IS RESPONSIBLE FOR CALCULATING THE QUALITY OF THE    QUAL        4
C     COMPLEX FIELD.                                                    QUAL        5
      LEVEL 2, CU,CUR,US                                                QUAL        6
      COMMON/MELT/CU(16384),CFIL(16512),X(128),WL,NPTS,NPY,DRX,DRY      QUAL        7
      DIMENSION    TITLE(1) , US(16384) , ANS(1) , CUR(32768)           QUAL        8
     X,FOM(5),P(6),XSAVE(128)                                           CYCLE9     17
      COMPLEX CU,CFIL,CONE,CJ,CZERO                                     QUAL        9
      EQUIVALENCE (CFIL(1),US( 1 )) , (CUR(1),CU(1))                    QUAL       10
C ***** SAVE FIELD                                                      QUAL       11
      NP=NPTS/2                                                         QUAL       12
      NOB=NPTS*NPY                                                      QUAL       13
      IF (ISAVE.LT.9) GO TO 212                                         QUAL       14
      READ(9) (CU(IZ),IZ=1,NOB),X,DRX,DRY                               QUAL       15
      REWIND 9                                                          QUAL       16
      GO TO 310                                                         QUAL       17
```

```
  212 IF (ISAVE.NE.1) GO TO 211                                          QUAL
      WRITE(7) (CU(IZ),IZ=1,NOB),X,DRX,DRY                               QUAL
      REWIND 7                                                           QUAL
  211 IF (ISAVE.NE.-1) GO TO 210                                         QUAL
      READ(9) (CU(IZ),IZ=1,NOB),X,DRX,DRY                                QUAL
      REWIND 9                                                           QUAL
  210 CONE=(1.0E0,0.0E0)                                                 QUAL
      PI=3.141593                                                        QUAL
C     CJ=(0.E0,1.E0)                                                     QUAL
      CZERO=(0.E0,0.E0)                                                  QUAL
      RK=2.*PI/WL                                                        QUAL
      AX=0.                                                              QUAL
      AY=0.                                                              QUAL
      DCAL=X(NPTS)-X(1)+X(2)-X(1)                                        QUAL
      IF (IPHASE.EQ.0)GO TO 50                                           QUAL
C     CORRECT LINEAR AND QUADRATIC COMPONENTS OF THE PHASE.              QUAL
      CALL TILT(AX,AY,RADIUS,IPHASE)                                     QUAL
      IF(IPHASE.LT.2)GO TO 50                                            QUAL
      BHALF=PI/(WL*RADIUS)                                               QUAL
      DO 65 J=1,NPY                                                      QUAL
      J1=(J-1)*NPTS                                                      QUAL
      YSQ = X(J) **2                                                     QUAL
      DO 65 I=1,NPTS                                                     QUAL
      IJ=I+J1                                                            QUAL
      IJ2 = 2 * IJ                                                       QUAL
      IJ2M1 = IJ2 - 1                                                    QUAL
      PHI = BHALF * (X(I)**2 + YSQ)                                      QUAL
      SINP = SIN(PHI)                                                    QUAL
      COSP = COS(PHI)                                                    QUAL
      CURS = CUR(IJ2M1)                                                  QUAL
      CUR(IJ2M1) =  CURS*COSP - CUR(IJ2)*SINP                            QUAL
   65 CUR(IJ2)    =   CURS*SINP + CUR(IJ2)*COSP                          QUAL
C  65 CU(IJ)=CU(IJ)*CEXP(CMPLX(0.,BHALF*(X(I)**2+X(J)**2)))              QUAL
   50 CONTINUE                                                           QUAL
C *** STRENGTH OF LENSE REQUIRED TO KEEP BEAM WITHIN 2.* RF AT FOCUS     QUAL
      F = DCAL*DB/(2.*RF*WL)                                             QUAL
      DX=X(2)-X(1)                                                       QUAL
      DXSQ=DX**2                                                         QUAL
      PT=0.                                                              QUAL
      IZ=0                                                               QUAL

C      APPLY LENSE TO COMPLEX FIELD                                      QUAL
      DO 4 M=1,NPY                                                       QUAL
      YSQ = X(M) **2                                                     QUAL
      DO 4 N=1,NPTS                                                      QUAL
      IZ=IZ+1                                                            QUAL
      IZ2 = 2 * IZ                                                       QUAL
      IZ2M1 = IZ2 - 1                                                    QUAL
      PHI = RK * (X(N)**2 + YSQ) /2. /F                                  QUAL
      SINP = SIN(PHI)                                                    QUAL
      COSP = COS(PHI)                                                    QUAL
      CURS = CUR(IZ2M1)                                                  QUAL
      CUR(IZ2M1) = CURS*COSP - CUR(IZ2)*SINP                             QUAL
      CUR(IZ2)    = CURS*SINP + CUR(IZ2)*COSP                            QUAL
C     CU( IZ)=CU( IZ)*CEXP(CJ*RK*(X(N)**2+X(M)**2)/2./F)                 QUAL
    4 PT = PT + CUR(IZ2M1)**2 + CUR(IZ2)**2                              QUAL
C   4 PT=PT+CU( IZ)*CONJG(CU( IZ))                                       QUAL
      IF ( PT .LE. 0.0) GO TO 200                                        QUAL
      PWSAVE = PT * DXSQ *  NPTS / NPY                                   QUAL
      DO 295 I=1,NPTS                                                    CYCLE9
      XSAVE(I) = X(I)                                                    CYCLE9
  295  CONTINUE                                                          CYCLE9
      DX2SVE=DX*DX                                                       CYCLE9
      DXSAVE= DX                                                         CYCLE9
      PTSAVE=PT                                                          CYCLE9
      DO 300 I=1,5                                                       CYCLE9
      D=(I-3)/10.                                                        CYCLE9
  300 FBM(I)  =  F*(1.+ D)                                               CYCLE9
      WRITE(1) (CU(IJ),IJ=1,NOB),X,DRX,DRY                               CYCLE9
      REWIND 1                                                           CYCLE9
      ISTEP=0                                                            CYCLE9
  325 ISTEP= ISTEP + 1                                                   CYCLE9
      PT=PTSAVE                                                          CYCLE9
```

```
      DX=DXSAVE                                                         CYCLE9    32
      DXSQ =DX2SVE                                                      CYCLE9    33
       DO 220 I =1,NPTS                                                 CYCLE9    34
      X(I) = XSAVE(I)                                                   CYCLE9    35
  220  CONTINUE                                                         CYCLE9    36
      IF(ISTEP .EQ. 6 ) GO TO 335                                       CYCLE9    37
      F = FBM(ISTEP)                                                    CYCLE9    38
      GO TO 340                                                         CYCLE9    39
  335 F=FOPT                                                            CYCLE9    40
  340 CONTINUE                                                          CYCLE9    41
      PWSAVK = PWSAVE/1000.                                             QUAL      76
      ZLD=F*WL/DB                                                       QUAL      77
      PI=PT*DXSQ/(ZLD*ZLD) * NPTS / NPY                                 QUAL      78
C     PROPAGATE TO THE FAR FIELD                                        QUAL      79
      CALL STEP  (F,1.0,  0.0,0.1,1,1,0,0.,0.,1,0)                      QUAL      80
C     CHANGE X TO FAR FIELD X                                           QUAL      81
      DO 11 I=1,NPTS                                                    QUAL      82
   11 X(I)=X(I)/ZLD                                                     QUAL      83
      DX=DX/ZLD                                                         QUAL      84
      DXSQ=DX*DX                                                        QUAL      85
      UMAX=0.                                                           QUAL      86
C     LOCATE PEAK INTENSITY IN FAR FIELD                                QUAL      87
  310 DO 61 J=1,NPY                                                     QUAL      88
      J1=(J-1)*NPTS                                                     QUAL      89
      DO 61 I=1,NPTS                                                    QUAL      90
       IZ=I+J1                                                          QUAL      91
       IZ2 = IZ * 2                                                     QUAL      92
       US(IZ) = CUR(IZ2-1)**2 + CUR(IZ2)**2                             QUAL      93
C      US(IZ )=CU(IZ )*CONJG(CU(IZ ))                                   QUAL      94
       IF (US(IZ).LT.UMAX) GO TO 61                                     QUAL      95
       XPEAK=X(I)                                                       QUAL      96
       YPEAK=X(J)                                                       QUAL      97
       UMAX=US(IZ )                                                     QUAL      98
   61  CONTINUE                                                         QUAL      99
       IF(NPTS.EQ.NPY)GO TO 63                                          QUAL     100
       DO 62 J=1,NPY                                                    QUAL     101
       JJ = NPTS+1-J                                                    QUAL     102
       J1=(J-1)*NPTS                                                    QUAL     103
       DO 62 I=1,NPTS                                                   QUAL     104
       IZ=I+J1                                                          QUAL     105
   62  US(I+(JJ-1)*NPTS) = US(IZ)                                       QUAL     106
   63  UMXK=UMAX/1000.                                                  QUAL     107
       UMXI=PWSAVE*PI*(DB/(WL*F))**2/4.0                                QUAL     108
C      STREHL INTENSITY                                                 QUAL     109
       STREHL=UMAX/UMXI                                                 QUAL     110
C      CALCULATE PERCENT OF FAR FIELD POWER WITHIN RB RADIUS OF IPEAK   QUAL     111
       CALL POWWOW(NPTS,DX,X,US,XPEAK,YPEAK,RB,PRR)                     QUAL     112
       PRR = PRR * ZLD**2                                               QUAL     113
       PRK = PRR/1000.                                                  QUAL     114
       P(ISTEP)=PRK                                                     CYCLE9    42
C      LOCATE INTENSITY CENTROID IN FAR FIELD                           QUAL     115
       CALL CENBAR ( NPTS, DX, X, US, XCINT, YCINT, UMAX )              QUAL     116
C      CALCULATE PERCENT OF FAR FIELD POWER WITHIN RB RADIUS OF CENTROID QUAL     117
       CALL POWWOW(NPTS,DX,X,US,XCINT,YCINT,RB,PRB)                     QUAL     118
       PRB = PRB * ZLD**2                                               QUAL     119
       PBK = PRB/1000.                                                  QUAL     120
       IF (ISTEP.EQ.6) GO TO 5904                                       CYCLE9    43
       IF (ISTEP.EQ.1) WRITE(6,5910)                                    CYCLE9    44
 5910 FORMAT (55X,19HFLUX IN 1RL/D ABOUT     /                          CYCLE9    45
     A 20H TRIAL FOCAL LENGTHS, 9X,16HTOTAL DCALC FLUX ,                CYCLE9    46
     X 9X,4HIMAX,9X,8HCENTROID  )                                       CYCLE9    47
       WRITE (6,5920) ISTEP,F,PWSAVK,PRK,PBK                            CYCLE9    48
 5920 FORMAT (3H  F,I1,1H=,G12.4,16X,F7.2,12X,F7.2,8X,F7.2)             CYCLE9    49
       GO TO 5930                                                       CYCLE9    50
 5904 WRITE(6,5940) F                                                   CYCLE9    51
 5940 FORMAT(22H OPTIMUM RESULTS AT F=,G12.4)                           CYCLE9    52
       WRITE(6,132) RB,PBK,XCINT,YCINT,RB,PRK,UMXK,XPEAK,YPEAK,PWSAVK,DB QUAL     121
  132 FORMAT(//15H DCALC FLUX IN ,F5.2,6H RL/D=,G12.4,27H ABOUT CENTROID QUAL     122
     X COORDINATES,2G12.4//15H DCALC FLUX IN ,F5.2,6H RL/D=,G12.4,14H AB QUAL     123
     XOUT IMAX OF ,G12.4,12H COORDINATES,2G12.4//18H TOTAL DCALC FLUX=, QUAL     124
     XG12.4,22H    REFERENCE DIAMETER=,F6.2)                            QUAL     125
```

```
      WRITE(6,133)STREHL                                               QUAL      126
  133 FORMAT(/19H STREHL INTENSITY =,G11.4)                            QUAL      127
 5930 CONTINUE                                                         CYCLE9     53
      IF(ISTEP.LE.5) GO TO 345                                         CYCLE9     54
      ANS(1) = HNB                                                     QUAL      128
      ANS(2) = PWSAVE                                                  QUAL      129
      ANS(3) = UMAX                                                    QUAL      130
      IF (HNB.GT.PHR) GO TO 53                                         QUAL      131
      XCINT = XPEAK                                                    QUAL      132
      YCINT = YPEAK                                                    QUAL      133
      ANS(1) = PHR                                                     QUAL      134
C     MAKE SPECIFIED KAR FIELD PLOTS AND CALCULATE POWER VS. R*LAMBDA/D QUAL      135
   53 IF (IPLT.NE.4) CALL PLTOT( NPTS, DX, X, UMAX, 4., DS, IPLT,      QUAL      136
     A TITLE,PI,XCINT,YCINT,DB,WL)                                     APR26      58
C ***** RESTORE FIELD                                                  QUAL      138
  345 CONTINUE                                                         CYCLE9     55
      IF(ISTEP.GE.6) GO TO 350                                         CYCLE9     56
      READ(1)  (CU(IJ),IJ=1,NOB),X,DRX,DRY                             CYCLE9     57
      REWIND 1                                                         CYCLE9     58
      IF(ISTEP.LT.5) GO TO 325                                         CYCLE9     59
      POPT=-100                                                        CYCLE9     60
      DO 375 I=1,5                                                     CYCLE9     61
      IF(P(I).LE.POPT) GO TO 375                                       CYCLE9     62
      POPT=P(I)                                                        CYCLE9     63
      FOPT=FBM(I)                                                      CYCLE9     64
  375  CONTINUE                                                        CYCLE9     65
      GO TO 325                                                        CYCLE9     66
  350 CONTINUE                                                         CYCLE9     67
      IF (ISAVE.NE.1) RETURN                                           QUAL      139
      READ(7)  (CU(IZ),IZ=1,NOB),X,DRX,DRY                             QUAL      140
      REWIND                                                           QUAL      141
      RETURN                                                           QUAL      142
  200 WRITE(6,201)                                                     QUAL      143
  201 FORMAT(30H0NO POWER IN BEAM - JOB KILLED)                        QUAL      144
      STOP                                                             QUAL      145
      END                                                              QUAL      146
```

24. SUBROUTINE REGAIN

Called from: GDL

Calls: BLUMIT, CPUTIM, FUHS, GAINXY, ISOCAV, SIMPGG, VINO

a. Purpose -- REGAIN is primarily a driver program to direct the recalculation of the cavity gain medium at the end of each iteration as shown by Figure 51. Subroutine REGAIN computer printouts follow Figure 51. The routine controls the type of kinetics calculation (numerical or analytical closed form), calculation of the FUHS effect on the medium density, generation of plots, and input/output of medium data on disk. Most of the control for this routine is read in from subroutine CAVITY.

b. Relevant formalism - The only formal calculations performed in REGAIN are the summation of cavity aerodynamics and FUHS effect induced optical path variations, and the averaging of newly calculated gain distribution with that of the previous iteration. A simple linear averaging or weighting algorithm is used:

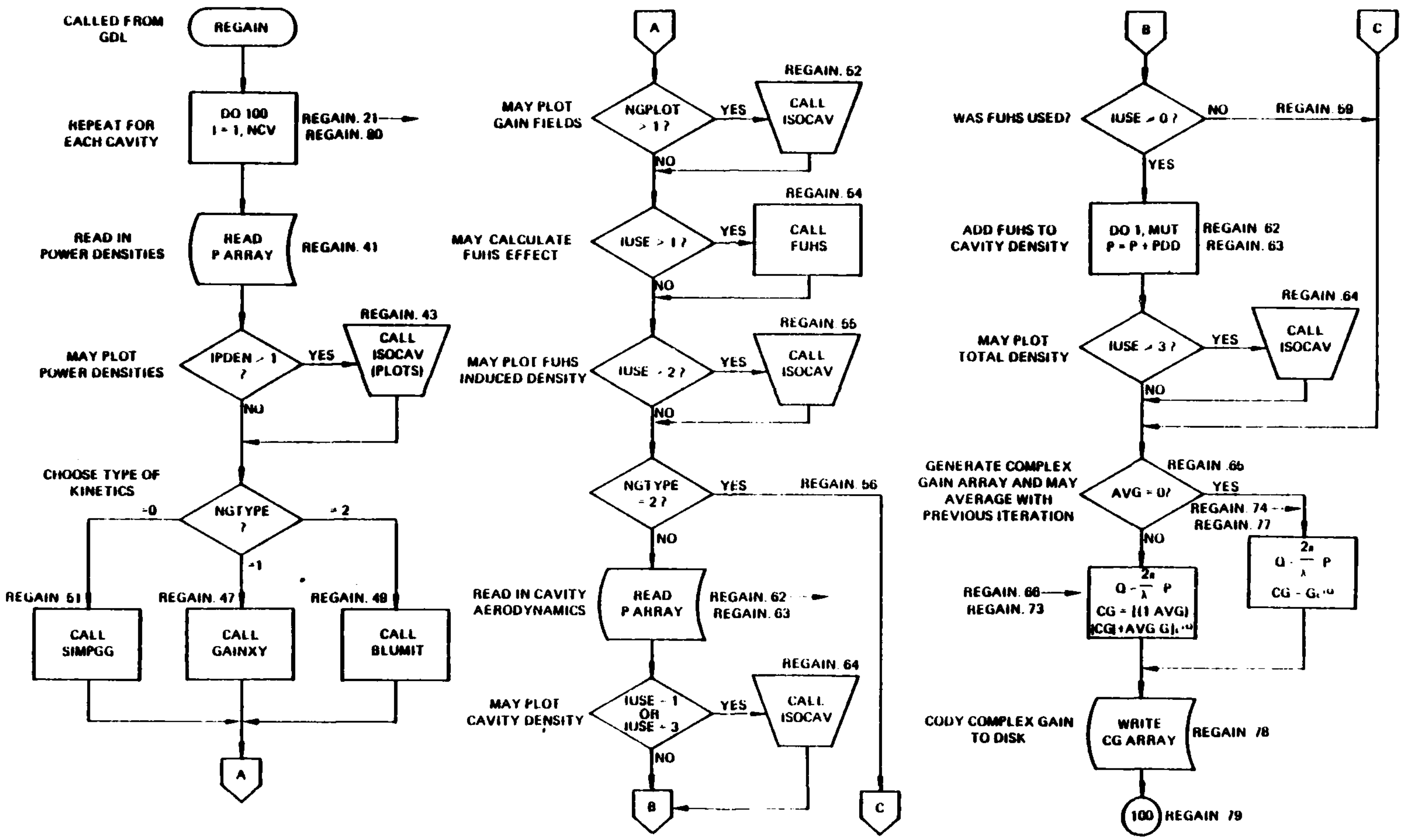


Figure 51. Subroutine REGAIN flow chart.

$$G = (G_o\ (1-A) + G_c A)\ \exp\left((2\pi/\lambda)\ OPD\right) \tag{153}$$

where

G_o is the amplitude gain field from the previous iteration,

G_c is the newly calculated amplitude gain field,

OPD is the sum of optical path differences,

λ is the wavelength.

Argument List

NCT	the number of cavity elements in the resonator
NIT	the iteration number

Commons Modified

/CCG/

Variables Modified

CG	the complex gain field

Relevant Variables

AVGG	weighting factor for averaging new and old gain arrays - defined by input to GDL
IBASE	integer reference number to control reading and writing power densities, gain, etc. to and from disk
IPDEN*	flag for plotting power densities
IUSE*	flag for FUHS calculation
NGPLOT*	flag for plotting gain fields
NGTYPE*	flag for controlling type of kinetics calculation
NSA*	number of gain/phase segments
NXA*	number of points in flow direction
NYA*	number of points across cavity (side-to-side)

*Defined by input to CAVITY

```
      SUBROUTINE REGAIN(NCT,NIT)                                          REGAIN     2
C     THIS ROUTINE DIRECTS (1) THE RECALCULATION OF GAIN AFTER A          REGAIN     3
C     RESONATOR ITERATION AND (2) THE GENERATION OF ANY SPECIFIED         REGAIN     4
C     PLOTS OF THE CAVITY PARAMETERS.                                     REGAIN     5
      LEVEL 2, CU,PDD,G,P,CG                                              REGAIN     6
      LEVEL 2,XC                                                          CORR2      9
      COMMON/MELT/CU(16384),CFIL(16512),X(128),WL,NPTS,NPY,DRX,DRY        REGAIN     7
      COMMON /CCG/ CG(17100)                                              C10DENS   32
      COMMON /GGGGG/ G(17100)                                             SOU77CY1 189
      COMMON/CAV2/  XC(5),YC(5),ZC(5),NX(5),NY(5),NS(5),XMC(5),YMC(5),    REGAIN     8
     1 NGTY(5),  NGPL(5),  IU(5),  IPD(5),                                REGAIN     9
     2                             SSGAIN(190,5),SATIN(5),BETA(5),RHOS(5), REGAIN   10
     3 VEL(5),GAM(5),XMACH(5),TV1(5),TV2(5),TV3(5),TVN2(5),TSCAV(5),      REGAIN    11
     4 PSCAV(5),PB(5),FN2(5),FCO2(5),FH2O(5),FCO(5),FO2(5),               REGAIN    12
     5 TITLE(20),AVG(5),NSYM                                              REGAIN    13
      DIMENSION PDD(16384),P(16384),G(16384)                              C10DENS   33
      COMPLEX CU,CFIL,CG,CAKAY                                            REGAIN    15
      EQUIVALENCE (PDD(1),CFIL(1)) ,                                      SOU77CY1 190
     X  (P(1),CU(1))                                                      C10DENS   34
      CALL CPUTIM(ISRT)                                                   REGAIN    18
C     CAKAY = CMPLX(0.,2.*3.141592/WL)                                    REGAIN    19
      TPIOL = 6.283184 / WL                                               REGAIN    20
      DO 100 NCV=1,NCT                                                    REGAIN    21
      IBASE = 10+(NCV-1)*11                                               REGAIN    22
      NGTYPE=NGTY(NCV)                                                    REGAIN    23
      NGPLOT=NGPL(NCV)                                                    REGAIN    24
      IUSE=IU(NCV)                                                        REGAIN    25
      IPDEN=IPD(NCV)                                                      REGAIN    26
      AVGG=AVG(NCV)                                                       REGAIN    27
      NSA=NS(NCV)                                                         REGAIN    28
      NXA=NX(NCV)                                                         REGAIN    29
      NYA=NY(NCV)/(NSYM+1)                                                REGAIN    30
      MUT=NXA*NYA                                                         REGAIN    31
      NEWC = 0                                                            REGAIN    32
      MMM  = 0                                                            REGAIN    33
      DO 90 L=1,NSA                                                       REGAIN    34
      IF (           NGPLOT.NE.-1 )  GO TO 18                             REGAIN    35
      NGPLOT = 3                                                          REGAIN    36
      IPDEN = 3                                                           REGAIN    37
      IUSE = 0                                                            REGAIN    38
      IF (IU(NCV).GE.1)  IUSE=3                                           REGAIN    39
   18 IPPP=IBASE+5+L                                                      REGAIN    40
      READ(IPPP) (P(IZ),IZ=1,MUT)                                         REGAIN    41
      REWIND IPPP                                                         REGAIN    42
      IF (IPDEN.GT.1) CALL ISOCAV(P, NCV, 2,  L  ,  NEWC,NIT,WL)          REGAIN    43
      IF (IPDEN.EQ.1.OR.IPDEN.EQ.3) CALL VINO(P,NCV, L ,NIT,2,MMM)        REGAIN    44
      ICC=IBASE+L                                                         REGAIN    45
C     CALL NUMERICAL GAIN ROUTINE                                         REGAIN    46
      IF (NGTYPE.EQ.1) CALL GAINXY(P,G,NCV,0)                             REGAIN    47
C     CALL MULTIBEAM THERMAL BLOOMING ROUTINE                             REGAIN    48
      IF (NGTYPE.EQ.2) CALL BLUMIT(P,G,NCV,WL)                            REGAIN    49
C     CALL CLOSED FORM GAIN ROUTINE                                       REGAIN    50
      IF (NGTYPE.EQ.0) CALL SIMPGG(P   ,G,NCV)                            REGAIN    51
      IF (  NGPLOT    .GE.2) CALL ISOCAV(G,NCV,1, L ,NEWC, NIT, WL)       REGAIN    52
      IF (NGPLOT.EQ.1.OR.NGPLOT.EQ.3) CALL VINO(G,NCV, L ,NIT,1,MMM)      REGAIN    53
      IF (IUSE.GE.1) CALL FORMS(P ,PDD,NCV)                               REGAIN    54
      IF (IUSE.GE.2) CALL ISOCAV(PDD, NCV, 3,  L  ,  NEWC, NIT, WL)       REGAIN    55
      IF (NGTYPE.EQ.2) GO TO 25                                           REGAIN    56
      READ (IBASE) (P(IZ),IZ=1,MUT)                                       REGAIN    57
      REWIND IBASE                                                        REGAIN    58
   30 IF (IUSE.EQ.-1)      GO TO 25                                       REGAIN    59
      IF (IUSE.EQ.0.OR.IUSE.EQ.3)     CALL ISOCAV(P,NCV,5,L,NEWC,NIT,WL)  REGAIN    60
      IF (IUSE.EQ.0)      GO TO 25                                        REGAIN    61
      DO 22 J = 1,MUT                                                     REGAIN    62
   22 P    (J) = P( J ) + PDD( J )                                        REGAIN    63
      IF (  IUSE .GE. 3  )  CALL ISOCAV(P,NCV,4, L ,NEWC,NIT,WL)          REGAIN    64
   25 IF (AVGG.EQ.0.) GO TO 21                                            REGAIN    65
      READ(ICC) (CG(IZ),IZ=1,MUT)                                         REGAIN    66
      REWIND ICC                                                          REGAIN    67
```

```
   20 DO 110 II=1,MUT
      PHI = P(II) * TPIOL
C 110 CG(II) =(G(II)*(1.-AVGG)+CABS(CG(II))*AVGG) * CEXP(CAKAY*P(II))
  110 CG(II) =(G(II)*(1.-AVGG)+CABS(CG(II))*AVGG) *
     X    CMPLX( COS(PHI) , SIN(PHI) )
      GO TO 23
   21 DO 112 II=1,MUT
      PHI = P(II) * TPIOL
C 112 CG(II)=G(II)*CEXP(CAKAY*P(II))
  112 CG(II)=G(II)*CMPLX(COS(PHI) , SIN(PHI) )
   23 WRITE(ICC) (CG(IZ),IZ=1,MUT)
   90 REWIND ICC
  100 CONTINUE
      WRITE(6,10)
   10 FORMAT(40H0GAIN HAS BEEN UPDATED FOR THE NEXT PASS)
      IF(NGTYPE .EQ. 1) WRITE(6,11)
   11 FORMAT(31H USING NUMERICAL KINETICS MODEL)
      IF(NGTYPE .EQ. 0) WRITE(6,12)
      IF(NGTYPE .EQ. 2) WRITE(6,19)
   19 FORMAT(34H USING THERMAL BLOOMING ANAYLYSIS )
   12 FORMAT(32H USING ANALYTICAL KINETICS MODEL)
      IF(IUSE .GT. 0) WRITE(6,13)
   13 FORMAT(70H0DENSITY VARIATIONS INDUCED BY LOWER LASER LEVEL RELAXAT
     XION CALCULATED)
      CALL CPUTIM(IFIN)
      DELT=(ISRT-IFIN)/100.
      WRITE(6,45) DELT
   45 FORMAT(1H0,G12.5,49H SECONDS OF CPU TIME SPENT IN SUBROUTINE REGAI
     XN  /1H1)
      RETURN
      END
```

SUBROUTINE RGRD   76/176   OPT=1   FTN 4.6+452   04/27/79   12.23.47

```
      SUBROUTINE RGRD(NRGD)
C
C       THIS ROUTINE REGRIDS CU FROM A NPTS**2 ARRAY TO AN
C       NRGD**2 ARRAY USING THE SAME GRID ELEMENT SIZE AS THE
C       ORIGINAL ARRAY.
C
      LEVEL 2, CU,CFILR
      COMMON/MELT/CU(16384),CFIL(16512),X(128),WL,NPTS,NPY,DRX,DRY
      DIMENSION CFILR(32768)
      COMPLEX CU,CFIL
      EQUIVALENCE (CFIL(1),CFILR(1))
      DX=X(2)-X(1)
      NFAC = NPTS/NPY
      NYAD=(NRGD-NPTS)/2
      NXAD=(NRGD-NPTS)/2
      X(1)=DX*(1-NRGD)/2.
      DO 10 I=2,NRGD
      X(I)=X(I-1)+DX
   10 CONTINUE
C     WRITE(6,101)X(1),X(NRGD)
C 101 FORMAT(//10X,6HX(1) =,G12.4,5X,9HX(NRGD) =,G12.4//)
C     CALL ZERO (CFIL(1),CFIL(16384))
      DO 173 IZERO=1,32768
  173 CFILR(IZERO)=0.
      DO 20 J=1,NPY
      INDX=NRGD*(NYAD+J-1)+NXAD
      NBASE=(J-1)*NPTS
      DO 30 I=1,NPTS
      CFIL(INDX+I)=CU(NBASE+I)
   30 CONTINUE
   20 CONTINUE
      NPTS=NRGD
      NPY = NPTS/NFAC
      NSQR=NPTS*NPY
      DO 40 IM=1,NSQR
      CU(IM)=CFIL(IM)
   40 CONTINUE
      RETURN
      END
```

25. SUBROUTINE RGRD

This routine regrids a complex amplitude field by adding zeroes to the array on all sides of the input field. Figure 52 is the flow chart for subroutine RGRD. Points added have the same separation as the original field. No interpolation or other formal calculation is necessary. Use of this routine has the effect of increasing the guard band around the field.

Argument List

NRGD	desired number of grid points across field

Relevant Variables

DX	separation of grid points before and after regridding
INDX	counter or index used to locate old grid within the new grid
NSQR	total numer of points in regridded field

Commons Modified

/MELT/

Variables Modified

CFIL	temporary field storage array
CU	complex amplitude field
NPTS	number of grid points in x-dimension
NPY	number of grid points in y-dimension
X	x-position array

26. SUBROUTINE ROSN

a. Purpose -- The purpose of subroutine ROSN is to provide an accurate and rapid numerical interpolation subprogram for the evaluation of cavity-induced density perturbations. The subroutine uses cubic spline processed data representing aerodynamically parameterized $\frac{\Delta\rho}{\rho}$ data to interpolate to the cavity mesh for the run in question as shown in the ROSN subroutine flow chart (Fig. 53). Subroutine ROSN requires that the user specify the relevant cubic spline coefficients and $\frac{\Delta\rho}{\rho}$ values. The subroutine calculates $\Delta\phi$ for an arbitrary cavity mesh point, (x,y).

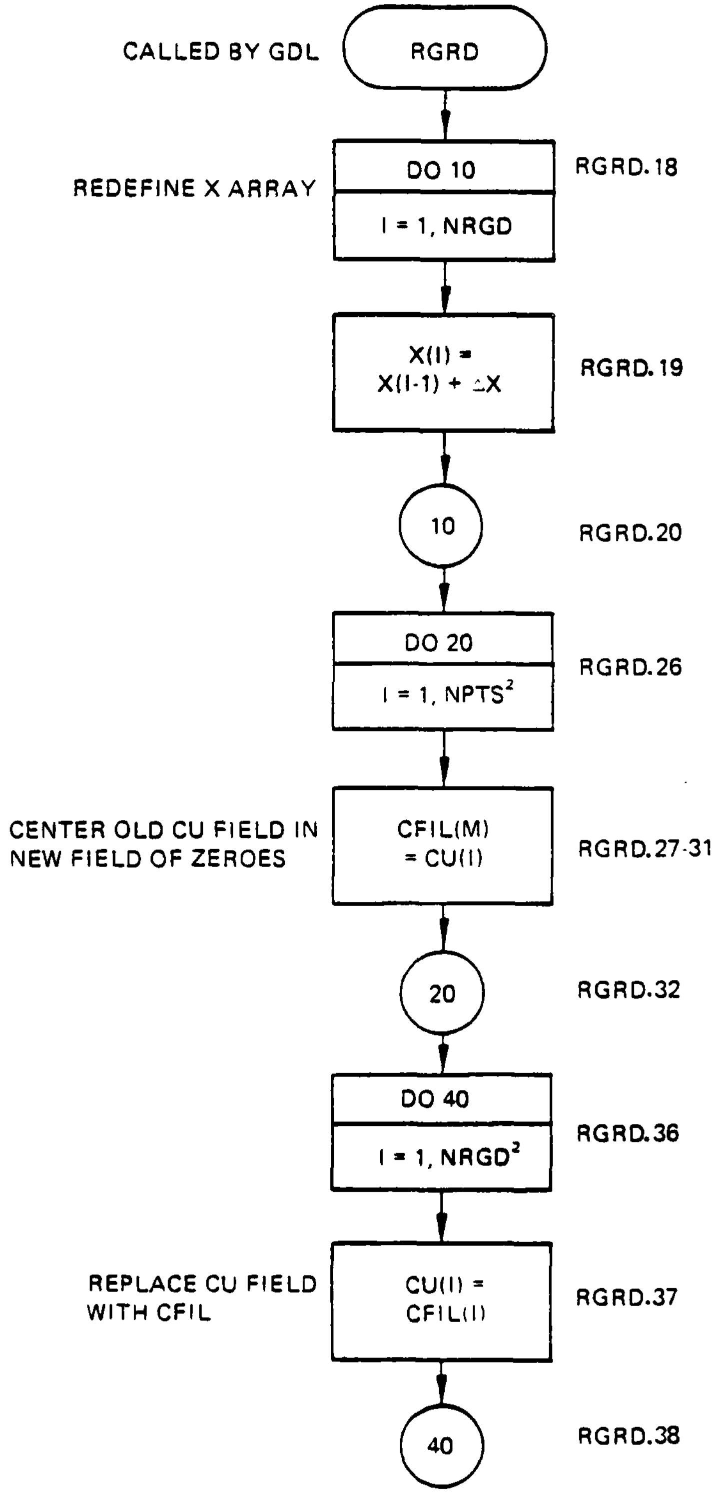
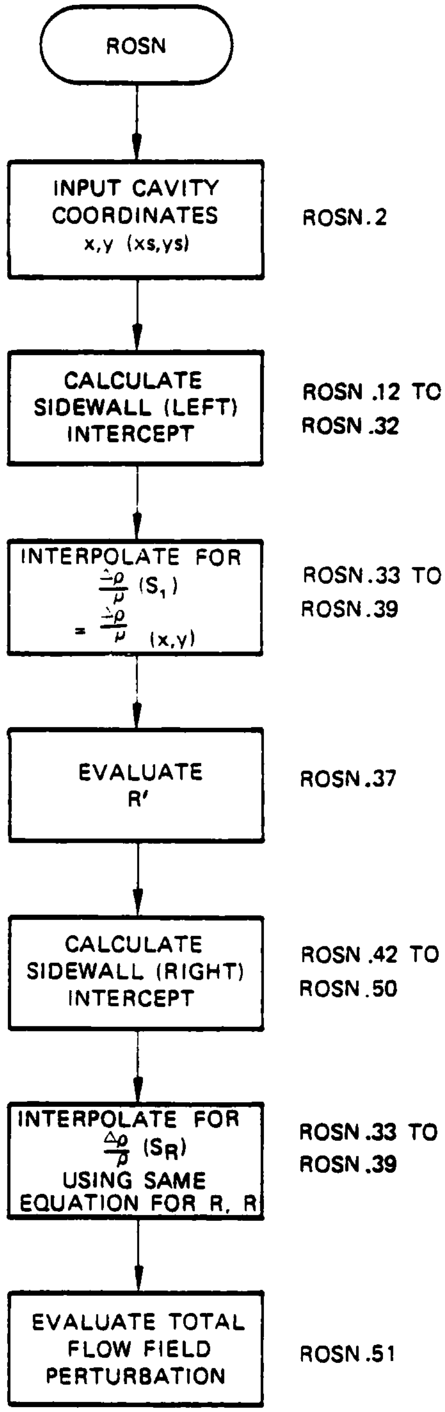


Figure 52. Subroutine RGRD flow chart.



$$S = \left[\frac{XLS(L)}{2} + XS\,\frac{YS}{TM(L)}\right] /W$$

$$R = D^*Z_{J+1} + E^*Z_J + \frac{H^2}{6}\left[C_J^*\,G(E) + C_{J+1}^*\,G(D)\right]$$

$$RS = \left[\frac{Z_{J+1} - Z_J}{H}\right] + \frac{H}{2}\left(C_{J+1}^*\,F(D) - C_J^*\,F(E)\right)$$

$$S = \left[\frac{XLS(L)}{2} + XS\,\frac{(W-YS)}{TM(L)}\right] /W$$

$$XNZ = CRHO(1)^*R1 + CRHO(L)^*R2$$

Figure 53. Subroutine ROSN flow chart.

b. Relevant formalism -- The SOQ Cavity coordinate system represents a regular mesh upon which many perturbations are applied. High Mach number flow produces ordered density gradients which may degrade beam phase relationships. Given arbitrary flow field interferometry it is possible to parameterize fringe shift ($\frac{\Delta\rho}{\rho}$ or ΔOPD) as a function of sidewall parameter s, where s is determined from the cavity sidewall projection of Mach lines, as shown in Figure 54.

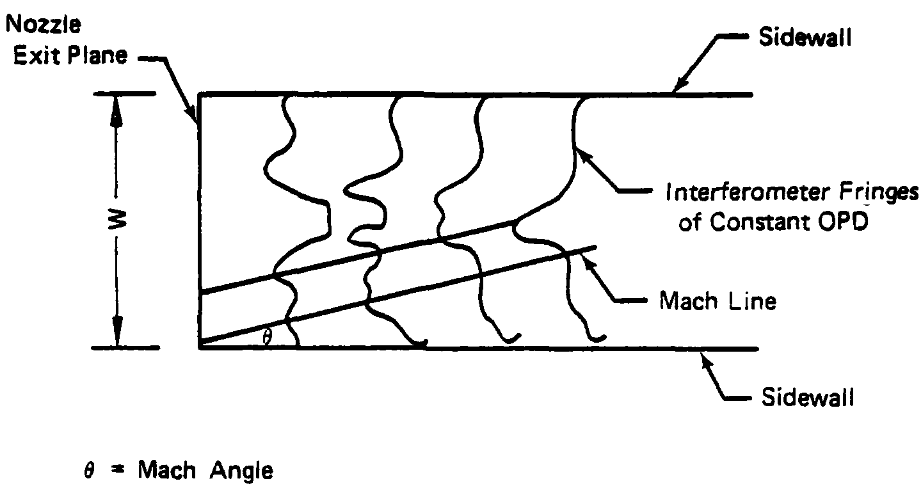


Figure 54. Fringe shift as a function of sidewall parameter.

From interferometry data and the above concept of sidewall projected data, the following parametric curves may be defined:

The curves shown in Figure 55 are fit using cubic splines, and the table or arrays of $\frac{\Delta\rho}{\rho} = f(s^*)$ and $C = g(s^*)$ (spline coeff) are stored in program DENSY. Subroutine ROSN is used to interpolate from (x,y) in the cavity to equivalent sidewall position s_{right} and s_{left} to determine using the above spline coefficients, an interpolated value of $\left.\frac{\Delta\rho}{\rho}\right|_{left} = f(s_{left}) = H(x,y)$

$$\left.\frac{\Delta\rho}{\rho}\right|_{right} = g(s_{right}) = K(x,y) \tag{154}$$

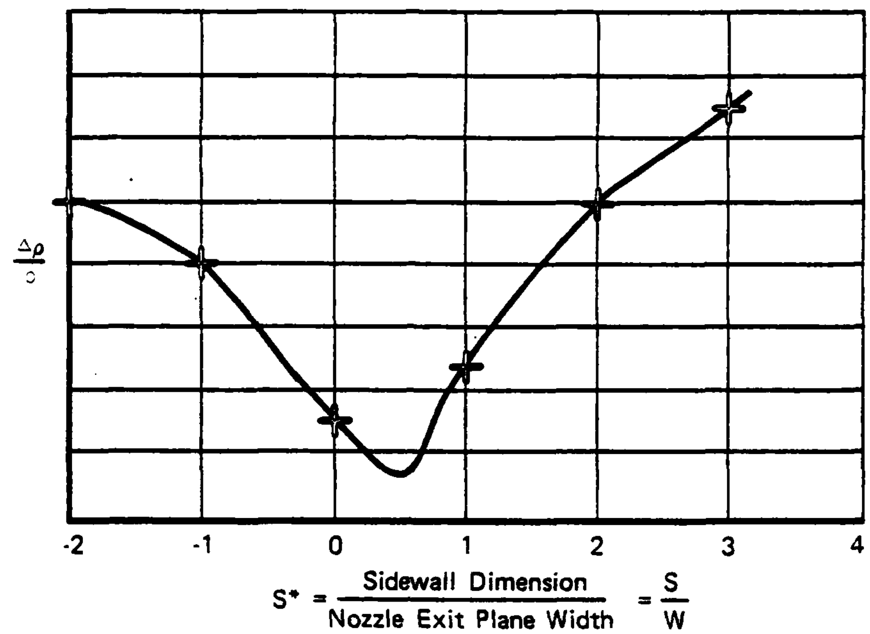


Figure 55. Parametric curves of Mach lines.

The $\left(\frac{\Delta\rho}{\rho}\right)$ at the point (x,y) is given, from supersonic flow theory as:

$$\left.\frac{\Delta\rho}{\rho_{CL}}\right|_{Total} = \left.\frac{\Delta\rho}{\rho_{CL}}\right|_{left} + \left.\frac{\Delta\rho}{\rho_{CL}}\right|_{right}$$

$$\Delta\phi = \frac{2\pi}{\lambda} C \left.\frac{\Delta\rho}{\rho_{CL}}\right|_{Total} \rho_{CL}$$

$$\Delta\phi = \Delta\varphi(x,y)$$

$$\left(\frac{\Delta\rho}{\rho}\right) = \frac{\Delta\rho\ (x,y)}{\rho}$$

The Spline interpolator is:

$$R = \frac{S^*-S_i}{S_{i+1}-S_i}\left(\frac{\Delta\rho}{\rho}\right)_{i+1} + \frac{S_{i+1}-S^*}{S_{i+1}-S_i}\left(\frac{\Delta\rho}{\rho}\right)_i + \left[\frac{(S_{i+1}-S_i)^2}{6}\right]$$
$$*\left\{[C_i]\left\langle\left(\frac{S_{i+1}-S^*}{S_{i+1}-S_i}\right) - \left(\frac{S_{i+1}-S^*}{S_{i+1}-S_i}\right)\right\rangle\right.$$
$$\left. + [C_{i+1}]\left\langle\left(\frac{S^*-S_i}{S_{i+1}-S_i}\right)^3 - \left(\frac{S^*-S_i}{S_{i+1}-S_i}\right)\right\rangle\right\} \tag{155}$$

The interpolator is evaluated for each of a right and left wall contribution along the appropriate Mach line.

Commons Modified

None

Commons Included

/LENSY/

Relevant Variables

XS	Position in cavity in cm along flow direction
XS	Position in cavity in cm orthogonal to flow direction
XNZ	Interpolated perturbation to flow field at (xs,ys)
S	Sidewall location
R	Interpolated density value
/LENSY/	
Y (51,2)	<-> abcissa y(51,1) <-> leftwall y(51,2) <-> right wall
Z (51,2)	<-> ordinates; same convention
C (51,2)	<-> Spline Coefficients; same convention
TM(2)	Tangent of Mach angle - left and right sides
XLS	Relative position of nep. read in subroutine densy.
W	cavity width (cm)
XMULT	scaling factor usually used to scale from % to absolute $\frac{\Delta\rho}{\rho}$
CRHO	Center line density left & right, may carry Gladstone-Dale constant
M(2)	number of left & right data points respectively
TITLE	Alphanumeric title
LL	No. of sidewall projections i.e., if left right symmetry is assumed, then LL=1, otherwise = 2.

```
      SUBROUTINE ROSN(XS,YS,XNZ)
C     CAVITY DENSITY FIELD INTERPOLATION ROUTINE
C     THIS ROUTINE USES SPLINE COEFFICIENTS TO INTERPOLATE THE CAVITY
C     DENSITY FIELD (DELTA RHO/RHO AND SPLINE COEFFICIENT VERSUS
C     SIDEWALL PARAMETERS ) ONTO THE CAVITY MESH.
      COMMON/DENSY/Y(51,2),Z(51,2),C(51,2),TM(2),XLS(2),W,
     X     XMULT(2),  CRHO(2),  M(2), TITLE(20),  LL
      DATA J/2/
      F(A)=A*A-1./3.
      G(A)=A*(A*A-1.)
      L = 1
      KY=M(L) -1
      MM = M(L)
      ITEST=0
      S=(XLS(L)/2.+XS-YS/TM(L))/W
    6 IF(S-Y(1,L))30,7,7
    7 IF(S-Y(MM,L))8,8,30
    8 IF(J-KY)20,20,9
    9 J=KY
   20 YD1=Y(J,L)-S
      YD2=Y(J+1,L)-S
      IF(YD1*YD2)5,5,22
   22 IF(YD1)10,10,23
   10 J=J+1
      IF(J-KY)20,11,11
   11 J=KY
      GO TO 5
   23 J=J-1
      IF(J)12,12,20
   12 J=1
    5 JP=J+1
      H=Y(JP,L)-Y(J,L)
      D=(S-Y(J,L))/H
      E=1.-D
      R=D*Z(JP,L)+E*Z(J,L)+H*H/6.*(C(J,L)*G(E)+C(JP,L)*G(D))
      RS=(Z(JP,L)-Z(J,L))/H+H/2.*(C(JP,L)*F(D)-C(J,L)*F(E))
      GO TO 31
   30 R=0.
      RS=0.
   31 IF(ITEST)32,32,33
   32 ITEST=1
      R1=R
      L = LL
      MM = M(L)
      KY = MM - 1
      RS1=RS
      S=(XLS(L)/2.+XS-(W-YS)/TM(L))/W
      J=MM-J
      GO TO 6
   33 XNZ=CRHO(1) * R1 + CRHO(L) * R
      RETURN
      END
```

27. SUBROUTINE LINTERP

a. Purpose -- This subroutine is used within the SOQ code to linearly interpolate sidewall projected $\frac{\Delta\rho}{\rho}$ cavity density information from sidewall projection to the cavity mesh. Data $\frac{\Delta\rho}{\rho}$ are stored in compressed form as univariate curves of $\frac{\Delta\rho}{\rho}$ versus sidewall projection parameters s, from which $\frac{\Delta\rho}{\rho}$ at any point in the GDL cavity may be obtained as shown in Figure 56.

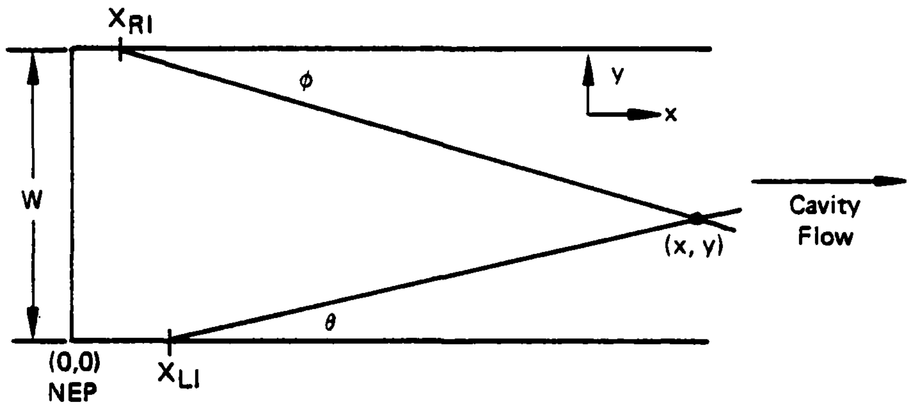


Figure 56. $\Delta\rho/\rho$ cavity density information.

The interpolated $\frac{\Delta\rho}{\rho}$ value is calculated to determine the equivalent flow-induced lens which is to be applied to the propagating wavefront. The lens is the result of flow-induced inhomogenieties such as ordered density gradients (weak shocks) and uneven thermal distribution.

The LINTERP subprogram (Fig. 57) calculates the sidewall parameters from interpolated cavity position (x,y) and Mach angle. With "s" determined for both right and left cavity sidewall projections a $\frac{\Delta\rho}{\rho}$ contribution can be determined for both sidewalls and linearly combined to give $\left(\frac{\Delta\rho}{\rho}\right)_{TOTAL} = f(x,y)$.

b. Relevant formalism

Left Intercept:

$$\tan\Theta = \frac{y}{(x-x_{LI})} \qquad x_{LI} = -\frac{y}{\tan\Theta} + x \tag{156}$$

where

(x,y) = interpolate position

X_{LI} = Left intercept

$\tan\Theta$ = tangent of Mach angle

sidewall parameter s

$$S_L = \frac{X_{LI}}{W} = \frac{(x - y/\tan\Theta)}{W}$$

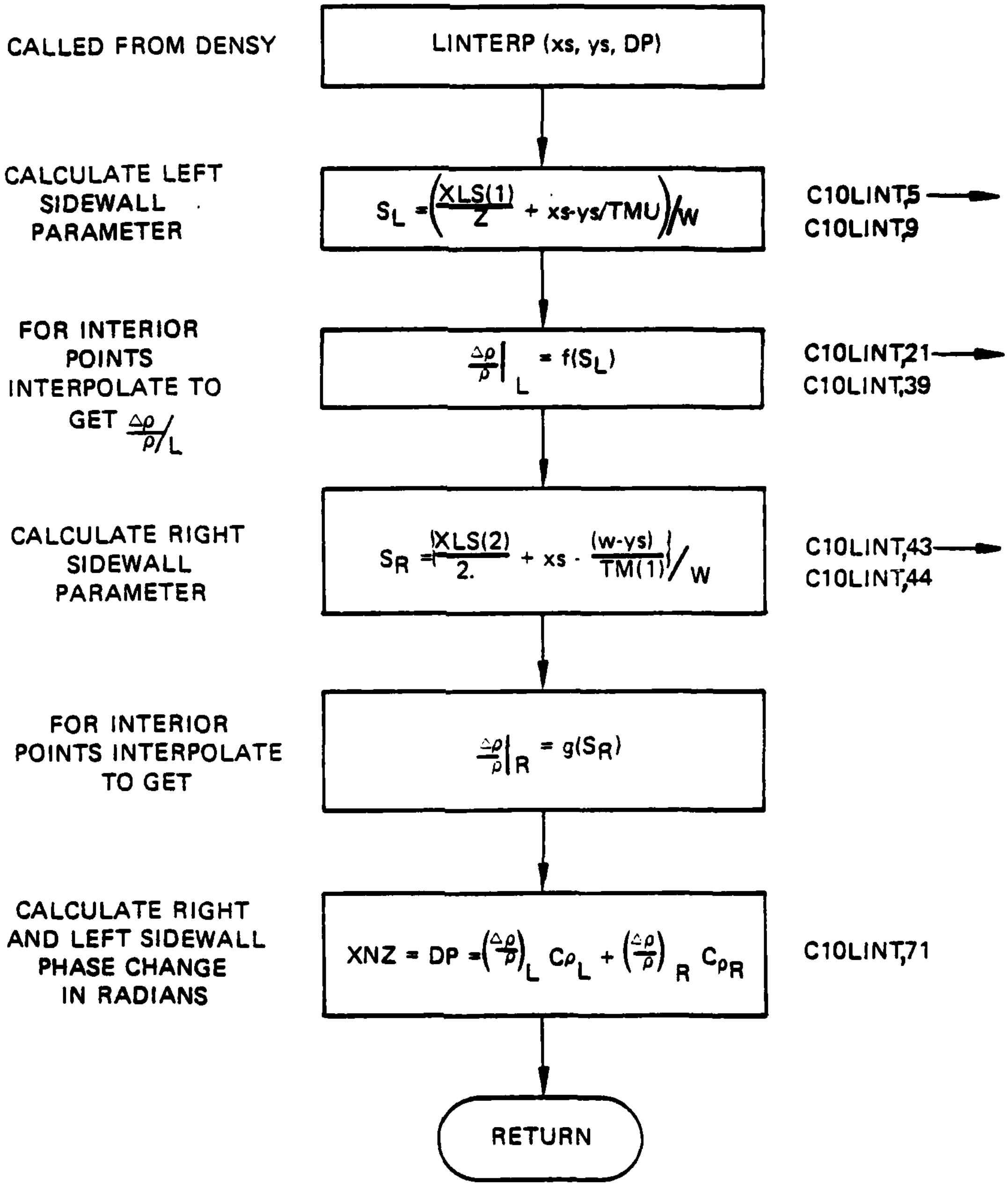


Figure 57. Subroutine LINTERP organization.

Right Intercept:

$$\tan\theta = \frac{w-y}{x-x_{R_I}} \tag{157}$$

$$\left(x - x_{R_I}\right)\tan\Theta = (w-y) - x\tan\Theta$$

$$X_{R_I} = \frac{-(w-y)}{\tan\Theta} + x$$

$$S_R = \frac{X_{R_I}}{W} = \frac{x-(w-y)/\tan\theta}{W} \tag{158}$$

where

w = cavity width

$\tan\Theta$ = tangent Mach angle

($\Theta \sim$ positive angle)

Commons modified

NONE

Definition of relevant variables

TM	Tangent of Mach angle
XLS	Arbitrary sidewall intercept offset (cm)
w	Cavity width (cm)
CRHO	Composite constant = $\frac{2\pi}{\lambda} C\Delta L\, \rho_o$

Subroutine LINTERP computer printouts follow.

SUBROUTINE LINTERP   76/176   OPT=1   FIN 4.6+452   04/27/79 12.23.47

```
      SUBROUTINE LINTERP(XS,YS,ANZ)
      COMMON/LENSY/Y(51,2),Z(51,2),C(51,2),TM(2),XLS(2),W,
     X             MULT(2),CHMU(2),M(2),TITLE(20),LL
C****** CALCULATE SIDEWALL PARAMETER  ********* ***************
      L=1
      MM=M(L)
      SL=(XLS(L)/2. + XS -YS/TM(L))/W
      IF(SL.LT.Y(1,L)) GO TO 5
      IF(SL.GE.Y(MM, L))GO TO 6
C ****** FIND S POSITION IN Y ARRAY*****
      DO 10 I = 1,MM
      IF(SL.GT.Y(I,L)) GO TO 10
      KL=I
      KLM1=I-1
      YSL=Y(I,L)
      YSLM1=Y(I-1,L)
      GO TO 15
   10 CONTINUE
   15 CONTINUE
```

```
C ******DETERMINE DRHO OVR RHOCL *******                          C1OLINT
C ****** FOR INTERIOR POINTS ***********                          C1OLINT
      YD1L=YSL - YSLM1                                            C1OLINT
      YD2L=SL - YSLM1                                             C1OLINT
      DRHO1= Z(KL,L) -Z(KLM1,L)                                   C1OLINT
      DRHO2= Z(KLM1,L)                                            C1OLINT
      LLL=1                                                       C1OLINT
C     IF(XS.GT.20.)                                               C1OLINT
C    XWRITE(6,92)KL,KLM1,Y(KL,L),Y(KLM1,L),Z(KL,L),Z(KLM1,L)      C1OLINT
   92 FORMAT(5X,*DO 10 LOOP*,2I5,4(5X,E15.7))                     C1OLINT
      DRHOL =(YD2L /YD1L)*DRHO1 + DRHO2                           C1OLINT
      GO TO 20                                                    C1OLINT
    5 DRHOL = Z(1,L)                                              C1OLINT
      LLL=2                                                       C1OLINT
      GO TO 20                                                    C1OLINT
    6 DRHOL = Z(MM,L)                                             C1OLINT
      LLL=3                                                       C1OLINT
   20 CONTINUE                                                    C1OLINT
C     IF(XS.GT.20.)WRITE(6,99)LLL,SL,DRHOL                        C1OLINT
   99 FORMAT(10X,I5,2(5X,E15.7),* LLL SL DRHOL*,/)                C1OLINT
C***** CALCULATE  SIDEWALL PARAMETER (RIGHT)******                C1OLINT
      L=LL                                                        C1OLINT
      MM= M(L)                                                    C1OLINT
      SR=(XLS(L)/2. + XS -(W-YS)/TM(L))/W                         C1OLINT
      IF(SR .LT.Y(1,L))GO TO 7                                    C1OLINT
      IF(SR .GE.Y(MM,L))GO TO 8                                   C1OLINT
      DO 40 I=1,MM                                                C1OLINT
      IF(SR.GT. Y(I,L) ) GO TO 40                                 C1OLINT
      KR=I                                                        C1OLINT
      KRM1= I - 1                                                 C1OLINT
      YDR1=Y(KR,L) - Y(KRM1,L)                                    C1OLINT
      YDR2= SR  - Y(KRM1,L)                                       C1OLINT
      GO TO 45                                                    C1OLINT
   40 CONTINUE                                                    C1OLINT
   45 CONTINUE                                                    C1OLINT
      DRHOR1= Z(KR,L) -Z(KRM1,L)                                  C1OLINT
      DRHOR2= Z(KRM1,L)                                           C1OLINT
      DRHOR=(YDR2/YDR1)*DRHOR1 + DRHOR2                           C1OLINT
      KKK=  1                                                     C1OLINT
C     IF(XS.GT.20.)                                               C1OLINT
C    XWRITE(6,93)KR,KRM1,Y(KR,L),Y(KRM1,L),Z(KR,L),Z(KRM1,L)      C1OLINT
   93 FORMAT(5X,*DO 40 LOOP*,2I5,4(5X,E15.7))                     C1OLINT
      GO TO 50                                                    C1OLINT
    7 DRHOR = Z(1,L)                                              C1OLINT
      KKK=2                                                       C1OLINT
      GO TO 50                                                    C1OLINT
    8 DRHOR = Z(MM,L)                                             C1OLINT
      KKK=3                                                       C1OLINT
   50 CONTINUE                                                    C1OLINT
C     IF(XS.GT.20.)WRITE(6,199)KKK,SR,DRHOR                       C1OLINT
  199 FORMAT(10X,I5,2(5X,E15.7),* KKK,SR,DRHOR*,)                 C1OLINT
      XNZ= DRHOL*CRHO(1) + DRHOR*CRHO(L)                          C1OLINT
C     IF(XS.GT.20.)WRITE(6,299) CRHO(1),CRHO(L)                   C1OLINT
  299 FORMAT(20X,*CRHO(1),CRHO(L)   *,2(E15.7),/)                 C1OLINT
      RETURN                                                      C1OLINT
      END                                                         C1OLINT
```

28. SUBROUTINE ROSN6

a. Purpose -- Subroutine ROSN6 (flow chart organization shown in Fig. 58) is incorporated into the SOQ code to allow inclusion of the cavity density field from direct interferogram data reduction. The data from interferometry are assumed to have been fit in the y (parallel to NEP) direction by cubic splines, using spaced points (not necessarily equal).

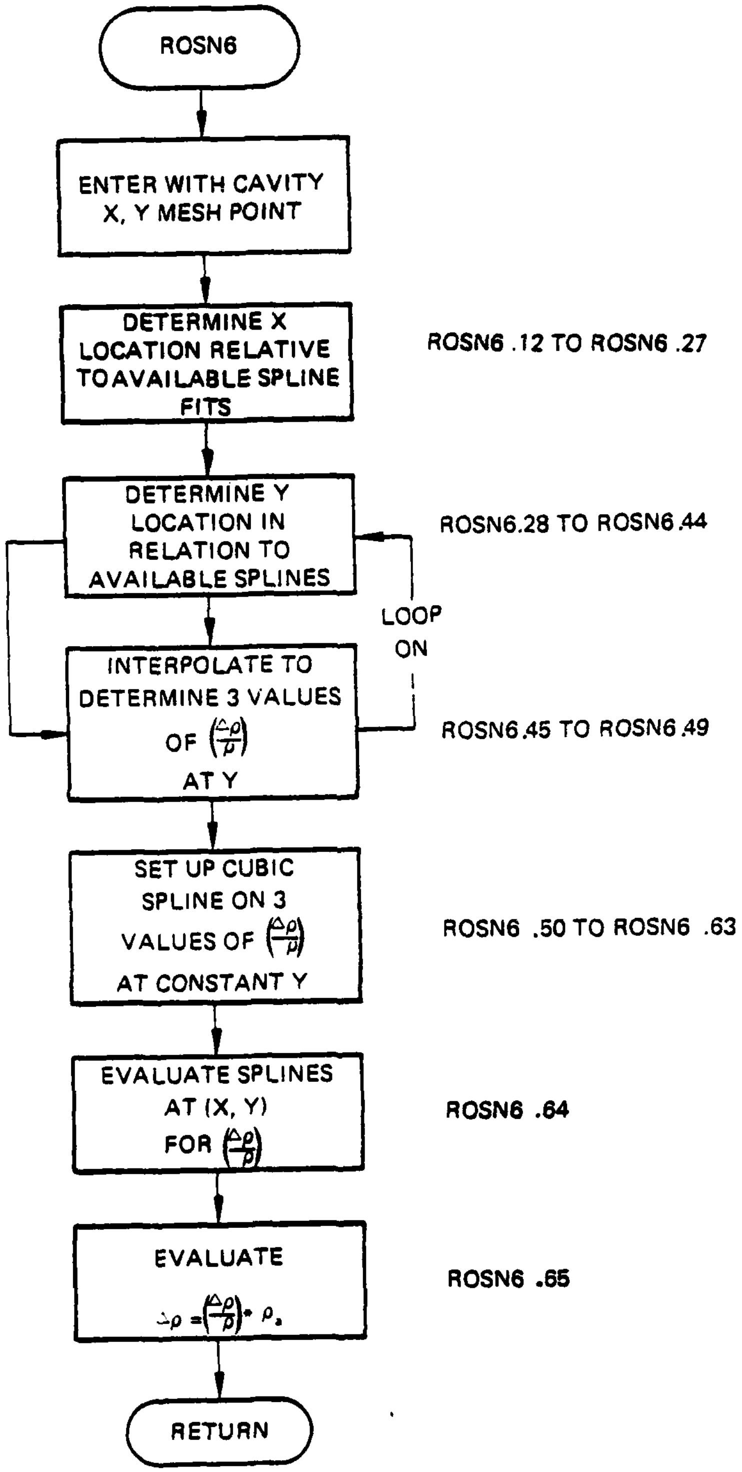


Figure 58. Subroutine ROSN6 organization.

Subroutine ROSN6 is a bivariate interpolation of the spline fit data using cubic splines.

b. Relevant formalism -- Subroutine ROSN6 uses the following procedure to interpolate the available spline data for an arbitrary cavity mesh point, (x,y), shown in Figure 59.

(1) Locate * in the spline fit data.
(2) Interpolate, using the spline fits at constant y, for the value of $\frac{\Delta\rho}{\rho}$ at the nearest three x values, (Δ).
(3) Construct a cubic spline in the direction (x_i, y^*) and evaluate at (x^*, y^*)
(4) Modify $\frac{\Delta\rho}{\rho_{CL}}$ (x^*, y^*) by $\frac{\Delta\rho}{\rho_{CL}}$ (x^*, y^*) to obtain $\Delta\rho$ in the desired units.

See page 214 for subroutine ROSN6 computer printouts.

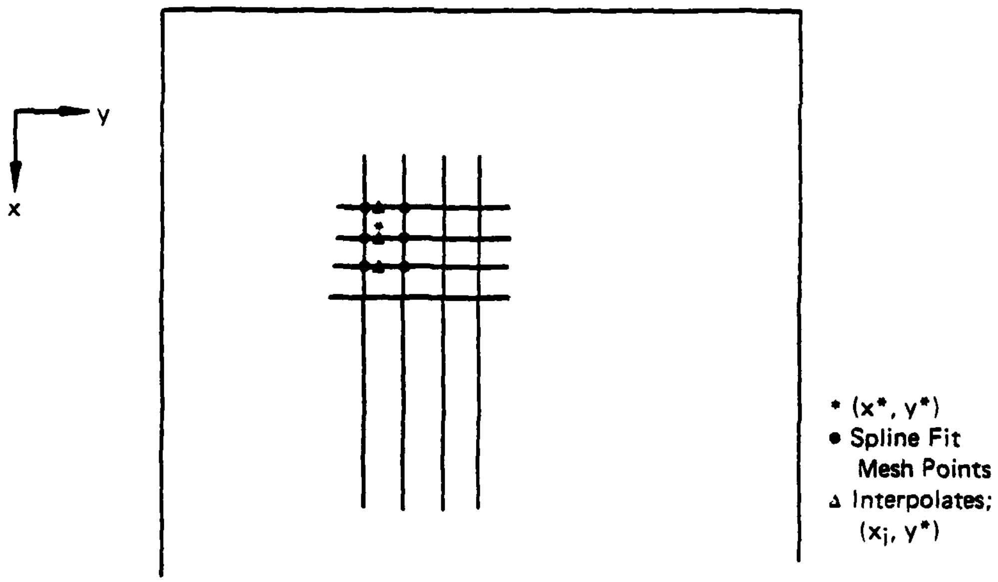


Figure 59. Available spline data for an arbitrary cavity mesh point.

Commons modified

/MELT/ not modified

/MELT/ is used to transfer in the following data:

x<=>cavity flow direction coordinates of spline fit data

y<=>orthogonal coordinates of spline coefficients

z<=>ordinate at each (x_i, y_j)

C<=>corresponding spline coefficients

M<=>Index array for constant x.

N

ROCL intended to be ρ at the center line but may be an artibrary scaling parameter.

Relevant Variables

xx cavity x-position

yy cavity y-position

XNZ ordinate interpolated at (x,y), normally $\Delta\rho = f(x,y)$

SUBROUTINE ROSN6 76/176 OPT=1 FIN 4.6+432 04/27/79 12.23.47

```
      SUBROUTINE ROSN6(XX,YY,XNZ)
C     THIS ROUTINE IS USED TO INTERPOLATE THE CAVITY DENSITY FIELD
C     (DELTA RHO/RHO AND SPLINE COEFFICIENT VERSUS X AND Y) ONTO THE
C     CAVITY MESH.
      LEVEL 2, PDUM
      COMMON / MELT /  PDUM(20000 ), X(21) ,
     X Y(21,81), Z(21,81),C(21,81),M(21),N,ROCL,DUMYS(40778)
      DIMENSION F(3),FP(3)
      DATA II,J/2,2/
      G(A)=A*(A*A-1.)
C COMPUTE LOCATION OF XX IN X(I)   X(1) .LE. XX .LE. X(N)
      KX=N-2
   10 XD1=X(II)-XX
      XD2=X(II+1)-XX
      IF(XD1*XD2)2,2,12
   12 IF(XD1 .GT. 0.) GO TO 13
      II = II+1
      IF(II .LT. KX) GO TO 10
      II=KX
      GO TO 2
   13 II = II-1
      IF(II .GT. 0) GO TO 10
      II =1
C COMPUTE THREE VALUES OF Z,AND DZ/DY AT YY
    2 L=II+2
      KK=0
C COMPUTE LOCATION OF YY IN Y(M(I))   Y(1) .LE. YY .LE. Y(M(I))
      DO 6 I=II,L
      KK=KK+1
      KY=M(I)-1
      IF(J .GT. KY) J=KY
```

```
20 YD1=Y(I,J)-YY
   YD2 = Y(I,J+1)-YY
   IF(YD1*YD2)5,5,22
22 IF(YD1 .GT. 0.) GO TO 23
   J=J+1
   IF(J .LT. KY) GO TO 20
   J=KY
   GO TO 5
23 J=J-1
   IF(J .GT. 0) GO TO 20
   J=1
 5 JP=J+1
   H=Y(I,JP)-Y(I,J)
   D=(YY-Y(I,J))/H
   E=1.-D
   F(KK)=D*Z(I,JP)+E*Z(I,J)+H*H/6.*(C(I,J)*G(E)+C(I,JP)*G(D))
 6 CONTINUE
C COMPUTE Z,DZ/DX,DZ/DY AT XX FROM CUBIC SPLINE THROUGH F AND FP
   H1=X(II+1)-X(II)
   H2=X(II+2)-X(II+1)
   IF(X(II+1)-XX)7,8,8
 7 D=(XX-X(II+1))/H2
   K=2
   H=H2
   GO TO 9
 8 D=(XX-X(II))/H1
   K=1
   H=H1
 9 E=1.-D
   CD=2.*((F(3)-F(2))/H2-(F(2)-F(1))/H1)/(H1+H2)
   TEM=H*H/6.*(G(E)+G(D))
   XN=D*F(K+1)+E*F(K)+CD*TEM
   XNZ=RDCL*XN
   RETURN
   END
```

29. SUBROUTINE SIMPGG

a. Purpose -- SIMPGG is used to calculate loaded gain for GDL cavities. It uses the E. A. Sziklas closed-form gain solution as derived in Reference 1, instead of numerically solving the appropriate GDL kinetics differential equations. SIMPGG also finds the intensity emitted at the gain/phase segment for use in FUHS. Figure 60 shows the SIMPGG organization.

b. Relative formalism -- The effect of the interaction of the light with the medium results in an amplification of the light beam as well as a phase change. Analytically this effect on the field is written

$$U (x,y) = t(x,y)U(x,y) \tag{159}$$

with

$$t(x,y) = \varepsilon^{\frac{1}{2}g(x,y)\Delta L}\ \varepsilon^{i\frac{2\pi}{\lambda}\Delta n\Delta L}$$

AD-A103 285 UNITED TECHNOLOGIES CORP WEST PALM BEACH FLA F/G 20/5
SYSTEM OPTICAL QUALITY USERS GUIDE. PART 2.(U)
MAR 80 J L FORGHAM, S S TOWNSEND F29601-77-C-0025
UNCLASSIFIED AFWL-TR-79-141-PT-2 NL

3 OF 3
AD A
103285

END
DATE
FILMED
10-81
DTIC

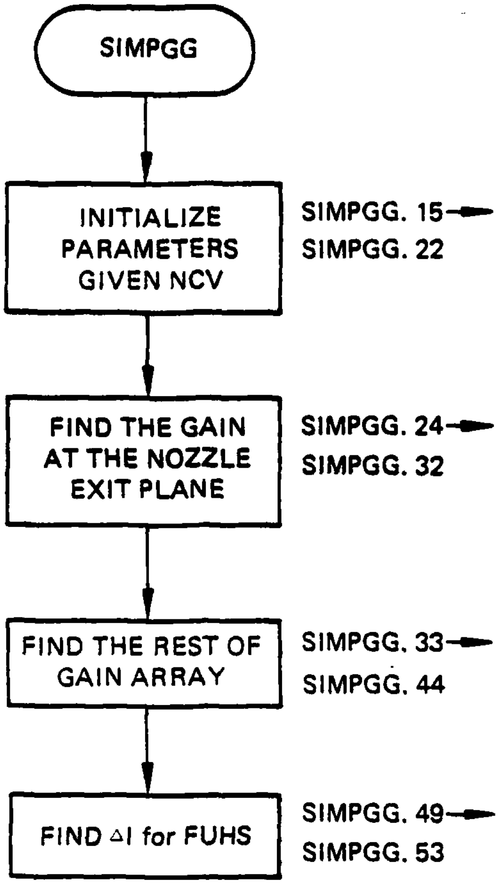


Figure 60. Subroutine SIMPGG organization.

ΔL is width of the medium under consideration, g (x,y) is the loaded gain coefficients and $\Delta n(x,y)$ is change in index of refraction due to density variations.

The factor of 1/2 in the exponent is due to the fact that gain is intensity, not amplitude, related:

$$I_{OUT} = I_{IN}\, e^{g\Delta L} = GI_{IN} \tag{160}$$

where

$$I = |U|^2$$

SIMPGG determines g(x,y) analytically using expression

$$g(x,y) = \left[\frac{g_o\ (x,y)}{1 + I(x,y)/I_{SAT}}\right] e^{\left(\frac{-X_{CO_2}\beta}{X_{N_2}\ V}\right)\int_{x_o}^{x} dx\ \frac{I(x,y)}{I_{SAT} + I(x,y)}} \tag{161}$$

and using the trapizoidal rule for the integral, where $g_o(x,y)$ is the small-signal gain coefficient found in subroutine GAINXY.

Note that

$$g(x,y)\Big|_{I(x,y) = 0} = g_o(x,y) \tag{162}$$

I_{sat} is the "saturation intensity"

$$I_{SAT} = \frac{h\upsilon\beta}{\sigma} \tag{163}$$

where

hυ is the photon energy, β the lower laser level relaxation rate, and σ the optical cross section for the transition. I_{sat} is also defined in subroutine GAINXY.

Where the FUHS routine is to be called to calculate heat increase in the gas due to lower level decay, the intensity change in the beam is needed for each gain phase segment, thus giving the heat release.

Consider Figure 61 of a gain/phase segment

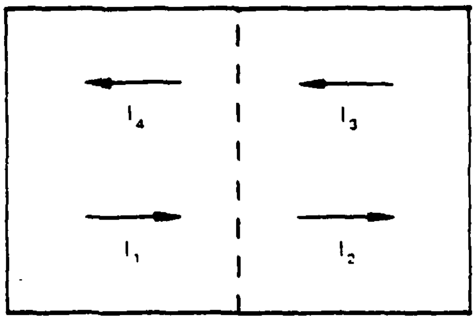


Figure 61. Gain/phase segment.

Then for each (I,J)

$$\Delta I = (I_1 + I_3) - (I_2 + I_4) \tag{164}$$

the quantity stored in the array PPD after a complete round trip is the average of the right running wave $(I_1 + I_2)/2$ plus the average of the left running wave $(I_3 + I_4)/2$.

Therefore

$$PPD = (I_1 + I_2 + I_3 + I_4)/2 \tag{165}$$

but $I_2 = GI$, and $I_4 = GI_3$

so $\Delta I = (1-G)\ (I_1+I_3)$

and $PPD = \left(\frac{1+G}{2}\right)(I_1+I_3)$

therefore

$$\Delta I = 2\left(\frac{1 - G}{1 + G}\right) * PPD \tag{166}$$

Knowing the total power change due to ΔI and the quantum efficiency η, the total heat released is found. The factor $\frac{1}{\Delta z}\left(\frac{1-\eta}{\eta}\right)$ is discussed in FUHS.

c. Fortran

Argument List

PPD = Total intensity (left running + right running waves) --

Becomes $\frac{1}{\Delta z}\left(\frac{1-\eta}{\eta}\right)\Delta I$ for use in FUHS

GG = Gain = $\ell^{-g\Delta z/2}$

NCV = cavity number

Commons modified -- none

Subroutines called - none.

Subroutine SIMPGG computer printouts follow.

SUBROUTINE SIMPGG 76/176 OPT=1 FTN 4.6+452 04/27/79 12.23.47

```
      SUBROUTINE SIMPGG (PPD,GG,NCV)
C     CLOSED FORM GAIN ALGORITHM
C     THIS ROUTINE USES THE E.A.SZIKLAS CLOSED FORM GAIN SOLUTION FOR
C     CO2 TO CALCULATE LOADED GAIN FOR THE GDL CAVITIES.
      LEVEL 2, XC,PPD,GG
      COMMON/CAV2/ XC(5),YC(5),ZC(5),NX(5),NY(5),NS(5),XMC(5),YMC(5),
     2 NGTYP(10),    IUS(10),    SSGAIN(190,5),SATIN(5),BETA(5),RHOS(5),
     3 VEL(5),GAM(5),XMACH(5),TV1(5),TV2(5),TV3(5),TVN2(5),TSCAV(5),
     4 PSCAV(5),PB(5),FN2(5),FCO2(5),FH2O(5),FCO(5),FO2(5),
     5 TITLE(20), AVG(5),NSYM
      DIMENSION    GG(1),             PPD( 16384),
     2                G(190),SGAINX(190),WINTS(190)
C     CALL CPUTIM(ISRT)
      NSA=NS(NCV)
      NYA=NY(NCV) / (NSYM+1)
      NXA=NX(NCV)
      SAT=SATIN(NCV)
      MUT= NXA*NYA
      DDXX= XC(NCV) / NXA
      ZXZ = ZC(NCV)/NS(NCV)/2.
      AC1=FCO2(NCV)*BETA(NCV)/FN2(NCV)/VEL(NCV)
C     WRITE(6,2) NSA,NYA,NXA,DDXX,ZXZ,AC1,(SSGAIN(K,NCV),K=1,NXA)
C   2 FORMAT(1H0,3I5,3G12.5/16(1X,8G12.5/))
      DO 80 J=1,NYA
      IZ=1+(J-1)*NXA
      POP  = PPD( IZ)/SAT
      POP1 = POP  + 1.
      SGAINX(J) = POP/POP1*DDXX/2.
      WINTS(J) = POP/POP1
      G(J) = SSGAIN(1,NCV)/POP1*EXP(-AC1*SGAINX(J))
   80 GG( IZ )=  EXP(G(J)*ZXZ)
      DO 110 I=2,NXA
C     WRITE(6,3) G(32),SGAINX(32),WINTS(32),GG(I-1,32)
C   3 FORMAT(1X,4G12.5)
      DO 110 J=1,NYA
      IZ = I+(J-1)*NXA
      POP  = PPD(IZ )/SAT
      POP1 = 1.+POP
      WINT= POP  /  POP1
      SGAINX(J) = SGAINX(J)+(WINT+WINTS(J))/2.*DDXX
      WINTS(J) = WINT
      G(J) = SSGAIN(I,NCV)    /POP1*EXP(-AC1*SGAINX(J))
  110 GG(IZ ) = EXP(G(J)*ZXZ)
      IF(IUS(NCV).LE. 0) GO TO 300
C
C     COMPUTE HEAT RELEASE FUNCTION FOR FOMS ANALYSIS
C
      ETA = .40
      HCONST=2.E+7*(1.-ETA)/ETA/(ZC(NCV)/NSA)
      DO 200 I=1,MUT
      BIGG=GG( I )**2
  200 PPD( I )=HCONST*PPD( I )*(BIGG-1.0)/(BIGG+1.0)
C 300 CALL CPUTIM(IFIN)
C     DELT=(ISRT-IFIN)/100.
C     WRITE(6,310) DELT
C 310 FORMAT(25H0 GAIN CALCULATIONS COST ,G12.5,20H SECONDS OF CPU TIME/
C    X/)
  300 RETURN
      END
```

The following is from Reference 1 and is included for the convenience of the reader.

The gain coefficient for a gas dynamic laser is decribed with the aid of a simple three-level model representing a flowing N_2-CO_2 system interacting with a 10.6μ beam. The relevant energy-level structure is illustrated schematically in Figure 62. The upper (001) and lower (100) laser levels of CO_2 are designated a and b, respectively. The symbols n_a and n_b denote the population densities occupying these levels. The first excited vibrational level of N_2 is nearly resonant with the upper laser level. The population density N is nearly resonant with the upper laser level. The population density N in this level preferentially pumps the upper laser level. Since the ground state CO_2 and N_2 populations, labelled n_o and N_o, are generally large compared to n_a, n_b, and N, the magnitudes of n_o and N_o, are relatively unaffected by transitions to and from the excited levels. Accordingly, n_o and N_o may be viewed as constants, i.e., $n_o/N_o = CO_2/x_{N_2}$ = constant where x_{CO_2} and x_{N_2} are the mole fractions of CO_2 and N_2.

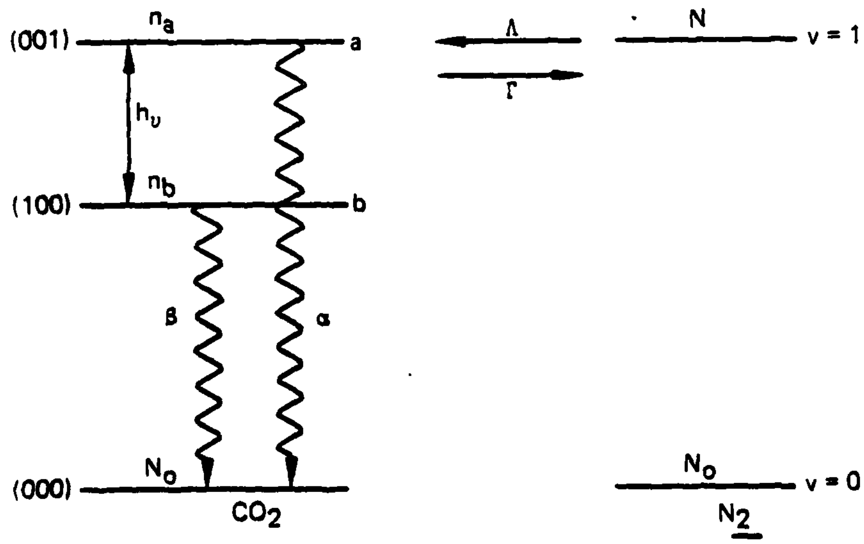


Figure 62. Relevant energy level diagram for N2-CO2 system.

For steady flow in the x-direction the rate equations describing the spatial variation of the three relevant population densities n_a, n_b and N are given by

$$v \frac{\delta n_a}{\delta x} = \Lambda N - (\alpha+\Gamma)n_a - (\sigma I/h\nu)\ (n_a - n_b) \tag{167}$$

$$v \frac{\delta n_b}{\delta x} = - \beta n_b = (\sigma I/h\nu)\ (n_a - n_b) \tag{168}$$

$$v \frac{\delta N}{\delta x} = \Gamma n_a - \Lambda N \tag{169}$$

Here, v is the flow velocity (assumed constant); α and β are the relaxation rates of the upper and lower levels; Λ and Γ are the forward and backward pumping rates of the upper laser level; σ is the optical cross section for the laser transition; $h\nu$ is the photon energy; and I is the beam intensity.

Since the pumping rates Λ and Γ are proportional to the ground state population densities n_o and N_o, respectively, it follows that

$$\Lambda/\Gamma = x_{CO_2}/x_{n_2} \tag{170}$$

Under typical GDL operating conditions $\chi_{CO_2} << \chi_{N_2}$. Also typically, the upper level decay rate is slow relative to the lower level decay rate, and the latter is slow relative to the backward pumping rate, i.e.,

$$\alpha << \beta,\ \Lambda << \Gamma \tag{171}$$

The beam is assumed to propagate in the z-direction. For purposes of analysis it is convenient to suppose that the transverse intensity profile at some axial station z can be divided into a series of constant intensity segments, as illustrated in Figure 63. For example, in the n^{th} segment $(x_n < x < x_{n+1})$ the intensity distribution is approximated by the value I_n = constant. For the moment, the segment width $x_{n+1} - x_n$ is left unspecified.

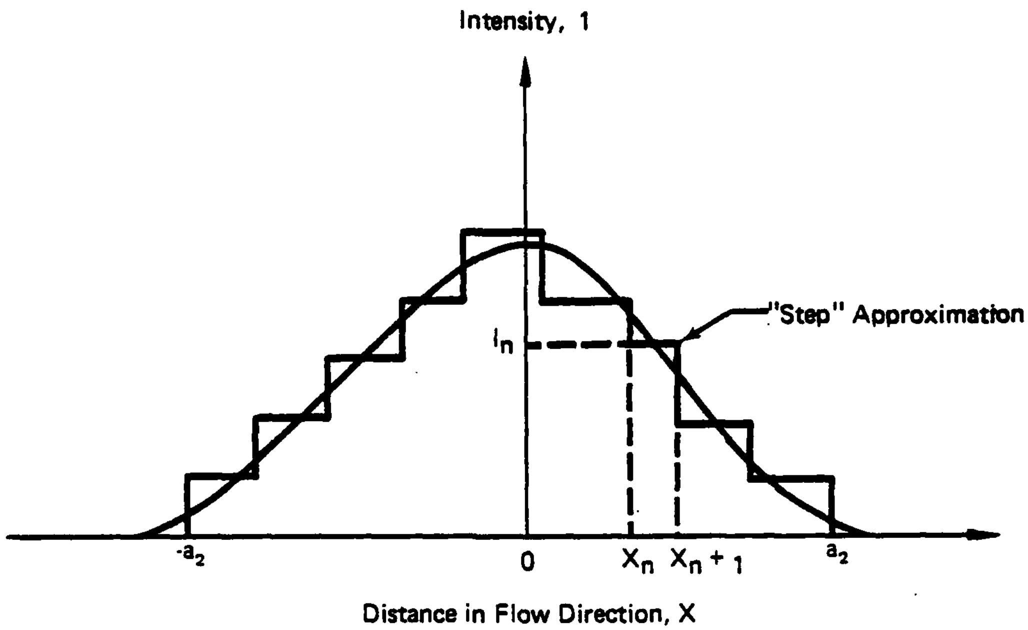


Figure 63. Step approximation to transverse intensity profile.

The gain coefficient for the laser transition is defined by

$$g(x,I) = \sigma(n_a - n_b) \tag{172}$$

We wish to solve for $g = g(x,I)$ in the n^{th} segment ($n = 1, 2, 3, \ldots$) where $I = I_n$ = constant. The upstream edge conditions $n_a(x_n)$, $n_b(x_n)$ and $N(x_n)$ are presumed known from the solution in the adjacent upstream segment. By successive application of the n^{th} segment solution, commencing with the segment at the upstream edge of the beam, one can in principle solve for g throughout the optical cavity.

The advantage of the segmented description is that an exact solution can be found in a region of constant beam intensity. Moreover, under suitable approximations, to be discussed later, this sequence of exact solutions can be put in a simple analytical form suitable for application to a smoothly varying beam profile.

Applying the Laplace transform to equations (167) through (169), one obtains

$$\underline{\underline{a}}\ \underline{b} = \underline{c} \tag{173}$$

where

$$\underline{\underline{a}} = \begin{Bmatrix} s+\alpha+\Gamma+W_n & -W & -\Lambda \\ -W_n & s+\beta+W_n & 0 \\ -\Gamma & 0 & s+\Lambda \end{Bmatrix}$$

$$\underline{b} = \begin{Bmatrix} \tilde{n}_a \\ \tilde{n}_b \\ \tilde{N} \end{Bmatrix} \qquad \underline{c} = \begin{Bmatrix} n_a(x_n) \\ n_b(x_n) \\ N(x_n) \end{Bmatrix}$$

Here, $\tilde{n}_a(s) = (1/v)\int_{x_n} dx\ n_a(x)\ \exp\left[-s(x - x_n)/v\right]$, etc.. and $W_n = \sigma I_n/h\nu$.

Solving by $\underline{b}$

$$n_a(s) = \left|det\right|^{-1}\left[(s+\delta+W_n)\ (s+\Lambda)n_a(x_n) + W_n(s+\Lambda)n_b(x_n) + \Lambda\ (s+\beta+W_n)N(x_n)\right] \tag{174}$$

$$n_b(s) = \left|det\right|^{-1}\left\{W_s(x+\Lambda)n_a(x_n) + \left[(s+\alpha+\Gamma+W_n)(s+\Lambda) - \Lambda\Gamma\right]n_b(x_n) + W_n\Lambda N(x_n)\right\} \tag{175}$$

$$N(s) = \left|det\right|^{-1}\left\{(s+\beta+W_n)\ \Gamma n_a(x_n) + W_n\Gamma n_b(x_n) + \left[(s+\alpha+\Gamma+W_n)(s+\beta+W_n) - W_n^2\right]N(x_n)\right\} \tag{176}$$

Here, $|det|$ is the determinant of a given by

$$\left|det\right| = s^3+k_2s^2+k_1s+k_o \tag{177}$$

where

$$k_2 \simeq \beta + \Lambda+\Gamma+2W_n$$

$$k_1 \simeq \beta(\Lambda+\Gamma) + W_n(2\Lambda+\Gamma+\beta)$$

$$k_o \simeq \Lambda\beta(\alpha+W_n)$$

The approximate equality sign refers to the use of the first half ($\alpha \ll \beta, \Lambda, \Gamma$) of the inequality 171.

Under the same approximation the roots of equation (177) are given by

$$r_1 \simeq \frac{\Lambda\beta(\alpha+W_n)}{\beta(\Lambda+\Gamma) + W_n\ (2\Lambda+\Gamma+\beta)} \tag{178}$$

$$r_2 = \frac{1}{2}\left[\Lambda+\Gamma+\beta+2W_n - \sqrt{(\Lambda+\Gamma-\beta)^2 + 4W_n\ (W_n-\Lambda)}\right] \tag{179}$$

$$r_3 = \frac{1}{2}\left[\Lambda+\Gamma+\beta+2W_n + \sqrt{(\Lambda+\Gamma-\beta)^2 + 4W_n\ (W_n-\Lambda)}\right] \tag{180}$$

where $|\det| = (s+r_1)\ (s+r_2)\ (s+r_3)$.

In the absence of a beam ($W_n = 0$) the roots r_1, r_2 and r_3 have a simple physical interpretation.

$$\begin{aligned} r_1 &\rightarrow r_1^o = \alpha\Lambda/(\Lambda+\Gamma) \\ r_2 &\rightarrow r_2^o = \beta \\ r_3 &\rightarrow r_3^o = \Lambda + \Gamma \end{aligned} \tag{181}$$

The value r_1^o defines the relaxation rate of the available laser energy (the upper laser level coupled to the vibrationally excited N_2) in the absence of a beam; r_2^o describes the lower level decay; and r_3^o is the rate at which pumping equilibrium between the excited CO_2 and N_2 is established. Typically, $r_1^o \ll r_2^o \ll r_3^o$.

As W_n is increased from zero, the physical identification of the roots r_1, r_2, and r_3 becomes somewhat obscure. However, the inequality $r_1 \ll r_2 \ll r_3$ appears to hold for all values of W_n. This feature leads to an important simplification.

*Care must be exercised not to introduce the second inequality at too early a stage in the calculation.

Taking the inverse Laplace transform of equations (174) through (176) one obtains a solution in the form

$$n_a(x) = A \exp\left[-r_1 (x-x_n)/v\right] + B \exp\left[-r_2(x-x_n)/v\right] + C \exp\left[-r_3(x-x_n)/v\right] \quad (182)$$

where A, B, and C are functions of the initial conditions $n_a(x_n)$, etc., and of the various rate constants. Similar expressions hold for $n_b(x)$ and $N(x)$.

In the absence of a beam ($W_n = 0$) this solution reduces to the simple form

$$n_a(x) = \frac{\Lambda}{\Lambda+\Gamma}\left[n_a(x_n)+N(x_n)\right] \exp\left[-r_1^{\circ} (x-x_n)/v\right] + \left[\frac{\Gamma n_a(x_n)-\Lambda N(x_n)}{\Lambda+\Gamma}\right] \exp\left[-r_3^{\circ} (x-x_n)/v\right] \quad (183)$$

$$n_b(x) = n_b(x_n) \exp\left[-r_2^{\circ} (x-x_n)/v\right] \quad (184)$$

$$N(x) = \frac{\Gamma}{\Lambda+\Gamma}\left[n_a(x_n) + N(x_n)\right] \exp\left[-r_1^{\circ} (x-x_n)/v\right] - \left[\frac{\Gamma n_a(x_n) - \Lambda N(x_n)}{\Lambda+\Gamma}\right] \exp\left[-r_3^{\circ} (x-x_n)/v\right] \quad (185)$$

The quantity $\left[n_a(x) + N(x)\right]$, describing the available laser energy, decays at the characteristic rate r_1^{o} while the quantity $\left[\Gamma n_a(x) - \Lambda N(x)\right]$, describing the departure from pumping equilibrium, decays at the rate r_3^{o}.

When the beam intensity I_n is nonvanishing, the details of the solution become rather cumbersome, and successive application of this solution to a series of adjacent beam segments would be a tedious task. Fortunately this complexity can be largely eliminated with the aid of two physically reasonable assumptions.

The first assumption is that the segment widths $\Delta x_n = x_{n+1} - x_n$ can be made somewhat larger than the characteristic lengths v/r_2 and v/r_3. In other words, the intensity distribution $I = I(x)$ is assumed to vary little over the characteristic lengths for lower level decay and pumping equilibrium. In this event the second and third terms in equation (182), evaluated at the downstream edge of the n^{th} segment, can be neglected.

If, in addition, the rate of stimulated emission W_n ($n = 1,2,3,\ldots$) is less than the pumping equilibrium rate $\Lambda + \Gamma$, it follows that pumping equilibrium can be assumed throughout the optical cavity, i.e.,

$$\Gamma n_a(x) \approx N(x) \tag{186}$$

Application of these approximations yields for the population difference between laser levels evaluated at the downstream edge of the n^{th} segment

$$n_a(x_{n+1}) - n_b(x_{n+1}) \approx \frac{\beta(\Lambda+\Gamma)n_a(x_n)}{\beta(\Lambda+\Gamma) + W_n(2\Lambda+\Gamma+\beta)} \exp\left[-r_1(W_n)\frac{\Delta X_n}{v}\right]$$

$$\approx \frac{\beta}{\beta+W_n} n_a(x_n) \exp\left[-r_1(W_n)\frac{\Delta x_n}{v}\right] \tag{187}$$

where, in the latter expression, use has been made of the second half of the inequality (171).

By a similar procedure one finds

$$n_a(x_n) \approx n_a(x_{n-1}) \exp\left[-r_1(W_{n-1})\frac{\Delta x_{n-1}}{v}\right] \tag{188}$$

Repeated substitution of equation (188) into (187) gives

$$n_a(x_{n+1}) - n_b(x_{n+1}) \approx \frac{\beta n_a(x_1)}{\beta+W_n} \exp\left\{-\left[r_1(W_n)\Delta x_n + r_1(W_{n-1})\Delta x_{n-1} + \ldots + r_1(W_o)\Delta x_o\right]/v\right\} \tag{189}$$

If the segment widths Δx_n($n = 0, 1, 2,\ldots$) are now viewed as "infinitesimals" equation (189) may be rewritten

$$n_a(x)-n_b(x) = \frac{n_a(x_o)}{1+w(x)} \exp\left[-\frac{1}{v}\int_{x_o}^{x} dx' r_1\right] \tag{190}$$

$$= \frac{n_a(x_o)\ \exp\left[-r_1^{\bullet}\ (x-x_o)/v\right]}{1+w(n)} \exp\left[-\frac{1}{v}\int_{xo}^{x} dx'(r_1-r_1^{\bullet})\right]$$

where $w(x) = \sigma I(x)/h\nu\beta$ and x_o defines a convenient reference station (e.g., the upstream edge of the beam).

Using the basic definition (172), the rate expressions (178) and (181), the identity (170), and the inequality (171), one finds on substitution into (190)

$$g(x) = \left[\frac{g_o(x)}{1+w(x)}\right] \exp\left\{-\frac{x_{CO_2}\beta}{x_{N_2}v}\int_{x_o}^{x} dx' \frac{w(x')}{1+w(x')}\right\} \tag{191}$$

where g_o is the small-signal gain coefficient given by

$$g_o(x) = g_o(x_o) \exp\left[-\frac{x_{CO_2}\alpha\ (x-x_o)}{x_{N_2}\ v}\right] \tag{192}$$

It is instructive to note the physical significance of various terms appearing in equations (191) and (192). The term in square brackets in equation (191) is analogous to the usual gain expression for a homogenenously broadened line in a nonflowing laser medium. Here, however, the small-signal gain coefficient (192) is not constant, but decays exponentially with distance downstream. The nondimensional intensity $w(x)$ measures the rate of simulated emission $\sigma I/h\nu$ relative to the decay rate β of the lower level. For a nonflowing laser the value $w = 1$ defines the saturation intensity of the medium.

The exponential factor in equation (191) represents a corrective term due to flow. The probability that an initially excited CO_2 molecule will remain excited after traversing a beam is dependent on the beam profile encountered by the molecule upstream of the point in question. This explains the presence of an integral over the upstream flowpath in equation (191).

In summary, a simple approximate expression has been derived for the gain coefficient in a flowing N_2-CO_2 system. The validity of this expression rests on two principal assumptions: (1) instantaneous pumping equilibrium is maintained throughout the optical cavity and (2) the beam intensity changes slowly over the characteristic distance for lower level decay. Although these conditions are not always satisfied in practice, particularly near the upstream edge of the beam, it is believed that even in these instances equation (191) provides a qualitatively accurate description of gain saturation in a GDL. The gain coefficient defined by equation (191) is then included in the complex transmission function

$$t = \exp\left[g(x,y;I) \quad \Delta L/2 + i\Delta\phi \ (x,y;I)\right] \tag{193}$$

to describe the effect of the medium gain throughout a segment of length ΔL. Here, $\Delta\phi$ represents a phase shift due to possible refractive index variations.

30. SUBROUTINE SLIVER

a. Purpose -- Subroutine SLIVER, shown in Figure 64, applies an annular aperture to the field. It can be centered anywhere in the mesh.

b. Relevant formalism -- The field is set to zero interior to the annular aperture. Mesh squares intersecting the aperture edge have the field linearly adjusted for the relative area intersected by the aperture edge.

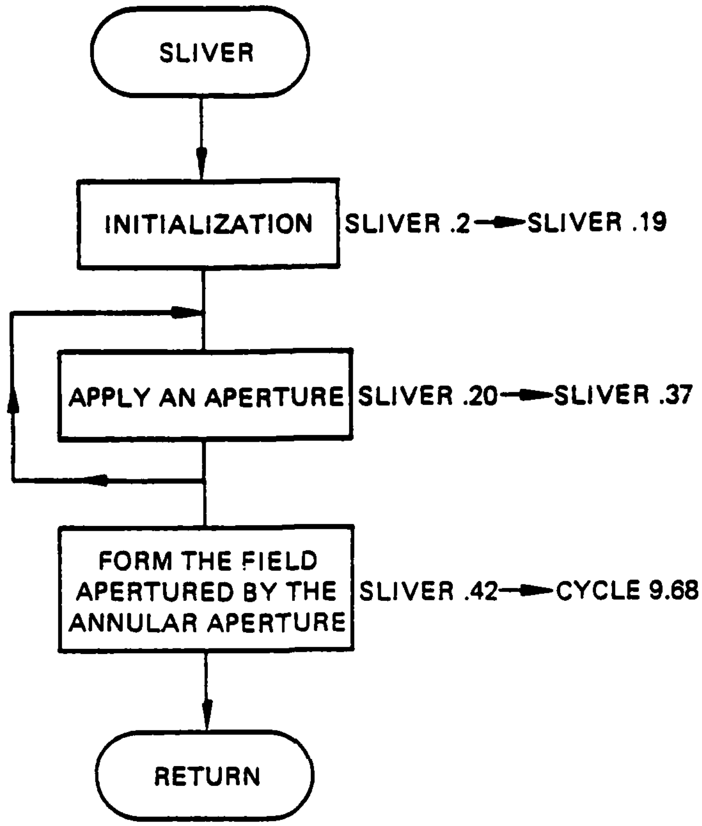


Figure 64. Subroutine SLIVER organization.

c. Fortran

Arguments

RIN = Radius of the OUTER edge of the annulus (cm)

ROUT = Radius of the INNER edge of the annulus (cm)

NOTE: Both RIN and ROUT must be negative to call "SLIVER" since if DOUT (=2*RIN) and DIN (=2*ROUT) are negative in the GDL call IFLOW = 4 section SLIVER is called instead of APRTR.

Common Variables Altered

CFIL = CFIL contains the original field

CU = CU is used to find the aperture field.

The Logic of Subroutine SLIVER is the following:

The final field is formed by subtracting an apertured field from the original. The aperture has a center disk of radius ROUT while the inner radius of the outer edge is RIN.

The center obscuration is first removed (IIN=0), then the outer obscuration (IIN=1). This apertured field (CU) is then subtracted from the original field (stored in CFIL) to form the field apertured by the annular aperture (CU).

The SLIVER subroutine computer printout follows.

SUBROUTINE SLIVER 76/176 OPT=1 FIN 4.6+452 04/27/79 12.23.47

```
      SUBROUTINE SLIVER(RIN,ROUT,XPOS,YPOS)                                 SLIVER    2
C     ANNULAR APERTURE TRANSMISSION FUNCTION                                SLIVER    3
C     THIS ROUTINE, WHICH OPERATES IN A MANNER SIMILAR TO SUBROUTINE        SLIVER    4
C     APERTURE, APPLIES AN ANNULAR OBSCURATION WITH INNER AND OUTER         SLIVER    5
C     RADII OF RIN AND ROUT, RESPECIVELY                                    SLIVER    6
      LEVEL 2, CU                                                           SLIVER    7
      COMMON/MELT/CU(16384),CFIL(16512),XAR(128),WL,NPTS,NPY,DRX,DRY        SLIVER    8
      COMPLEX CU,CFIL                                                       SLIVER    9
      RD(XX,YY,IX,IY)=SQRT((ABS(XX)+IX*DX/2.)**2+(ABS(YY)+DY/2.*IY)**2)     SLIVER   10
      RAPRTR=ABS(ROUT)                                                      SLIVER   11
      RDISK=ABS(RIN)                                                        SLIVER   12
      DX=XAR(2)-XAR(1)                                                      SLIVER   13
      DY=DX                                                                 SLIVER   14
      IIN=0                                                                 SLIVER   15
      RAD=RAPRTR                                                            SLIVER   16
      NOB=NPTS*NPY                                                          SLIVER   17
      DO 98 I=1,NOB                                                         SLIVER   18
   98 CFIL(I)=CU(I)                                                         SLIVER   19
   99 DO 101 IIX=1,NPTS                                                     SLIVER   20
      X=XAR(IIX)+DRX-XPOS                                                   SLIVER   21
      DO 101 IIY=1,NPY                                                      SLIVER   22
      Y=XAR(IIY)+DRY-YPOS                                                   SLIVER   23
      RPP=RD(X,Y,1,1)                                                       SLIVER   24
      RMM=RD(X,Y,-1,-1)                                                     SLIVER   25
      RMP=RD(X,Y,-1,1)                                                      SLIVER   26
      RPM=RD(X,Y,1,-1)                                                      SLIVER   27
      PER=1.                                                                SLIVER   28
      RMAX=AMAX1(RPP,RMM,RMP,RPM)                                           SLIVER   29
      IF (RMAX.LE.RAD) GO TO 100                                            SLIVER   30
      PER=0.                                                                SLIVER   31
      RMIN=AMIN1(RPP,RMM,RMP,RPM)                                           SLIVER   32
      IF (RMIN.GE.RAD) GO TO 100                                            SLIVER   33
      PER=(RAD-RMIN)/(RMAX-RMIN)                                            SLIVER   34
  100 IF (IIN.EQ.1) PER=1.-PER                                              SLIVER   35
      NNN = IIX+(IIY-1)*NPTS                                                SLIVER   36
  101 CU(NNN) = CU(NNN) * (1.-SQRT(PER))                                    SLIVER   37
      IF (RDISK.EQ.0..OR.IIN.EQ.1) GO TO 102                                SLIVER   38
      IIN=1                                                                 SLIVER   39
      RAD=RDISK                                                             SLIVER   40
      GO TO 99                                                              SLIVER   41
  102 DO 103 I=1,NOB                                                        SLIVER   42
      CU(I) = CFIL(I)-CU(I)                                                 CYCLE9   68
  103 CONTINUE                                                              CYCLE9   69
      WRITE(6,300) RAPRTR,RDISK                                             CYCLE9   70
  300 FORMAT (//28H ANNULAR OBSCURATION APPLIED /15H INSIDE RADIUS=,        CYCLE9   71
     X F10.3,17H  OUTSIDE RADIUS=,F10.3 )                                   CYCLE9   72
      RETURN                                                                SLIVER   44
      END                                                                   SLIVER   45
```

31. SUBROUTINE SPIDER

a. Purpose -- The SPIDER subroutine shown in Figure 65 applies an obscuration to the complex amplitude field in the form of several support struts, such as those used in a Cassegrain telescope system. Up to six struts at separate angles may be modeled. The result of the obscuration is listed in the output stream as an aperture loss.

b. Relevant formalism -- An angular deviation limit α calculated from the obscuration inside diameter d, the grid spacing Δx, and the strut width w, according to

$$\alpha = \sin^{-1} (w+2\Delta x)/d \tag{194}$$

Field points whose inclination angle is not within $\pm\alpha$ of a strut angle are assumed to be unobscured. Those points falling within this limit are subjected to closer inspection.

The distance δ from a grid center (x,y) to the strut centerline is calculated by

$$\delta = \left| y \cos\theta - X \sin\theta \right| \tag{195}$$

where Θ is the strut angle. The half-width of a grid measured along a normal to the strut h is calculated by

$$h = x/2./\mathrm{AMAX}\ (|\sin\theta|, |\cos\theta|) \tag{196}$$

then the maximum and minimum distance of the grid area from the centerline, d_{max} and d_{min} are

$$d_{max} = \delta + h$$

$$d_{min} = \delta - h$$

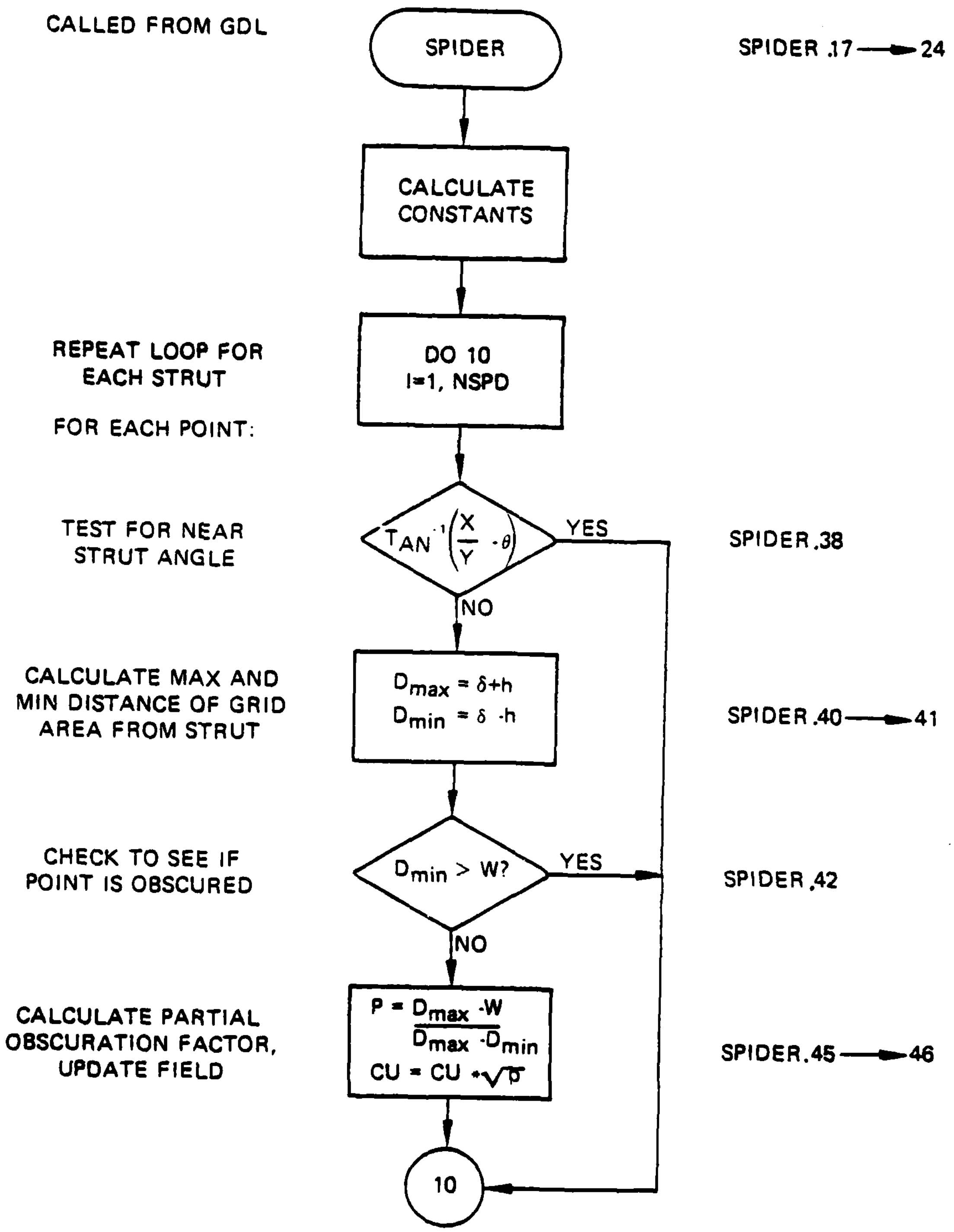


Figure 65. Subroutine SPIDER flow chart.

Points where d_{min} is greater than the strut half width h_s are not obscured. Points where d_{max} is less than the strut half width are totally obscured. The intensity of all other points is weighted according to

$$\text{intensity weighting} = (d_{max}-h_s)/(d_{max}-d_{min}) \tag{197}$$

Argument List

DIH	diameter of inner edge of support (hub)
NSPD	number of struts or spokes
THETA	array of strut angles
WIDTH	strut width
XC	x-position of center of obscuration
YC	y-position of center of obscuration

Relevant Variables

ANG	inclination angle of a point (x,y)
ANGTOL	angular width about the strut angle which defines the region to be searched for possible obscuration
DELTA	distance from (x,y) to the strut along a normal
DELXDH	half-width of coordinate grid measured along a normal to a strut
PER	weighting factor in establishing fractional obscuration

Commons Modified

/MELT/

CU the complex amplitude field.

The SPIDER subroutine computer printout follows.

SUBROUTINE SPIDER 76/176 OPT=1 FIN 4.6+452 04/27/79 12.23.47

```
      SUBROUTINE SPIDER (WIDTH,THETA,NSPD,XC,YC,DIH)                    SPIDER       2
C     GENERAL SUPPORT STRUT MODEL                                       SPIDER       3
C **** MODIFIED 10/17/75 TO HANDLE MULTIPLE THETAS ****                 SPIDER       4
C     THIS ROUTINE APPLIES AN OBSCURING STRUT TRANSMISSION FUNCTION TO  SPIDER       5
C     THE COMPLEX FIELD.  THE STRUT IS WIDTH WIDE WITH AN ANGLE THETA   SPIDER       6
C     (IN THE BEAM COORDINATE SYSTEM) AND GOES RADIALLY OUTWARD FROM    SPIDER       7
C     LOCATION (XC,YC). DIH IS HUB DIAMETER,NSPD IS NO. OF STRUTS.      SPIDER       8
C     DELXDH IS WIDTH/2 OF COORDINATE GRID ALONG NORMAL TO STRUT.       SPIDER       9
C     DELTA IS DISTANCE FROM X,Y TO CENTER OF STRUT ALONG NORMAL        SPIDER      10
C     TO STRUT.                                                         SPIDER      11
```

```
      LEVEL 2, CU
      COMMON/MELT/CU(16384),CFIL(16512),X(128),WL,NPTS,NPY,DNX,DNY
      DIMENSION THETA(1),THET(6),SINT(6),COST(6),DELXDM(6)
      COMPLEX CU,CFIL
      DATA PI,TWOPI/ 3.141593 ,6.283186 /
      WDTHM = WIDTH/2.0
      DELX02 = (X(2) -X(1)) / 2.
      ANGTOL = ASIN ((WIDTH+2.*(X(2)-X(1)))/ DIM )
      DO 5 IT=1,NSPD
      THET(IT) = THETA(IT)/57.3
      SINT(IT) = SIN(THET(IT))
      COST(IT) = COS(THET(IT))
    5 DELXDM(IT) = DELX02    / AMAX1(ABS(COST(IT)),ABS(SINT(IT)))
      IZ=0
      DO 10 J=1,NPY
      DO 10 I=1,NPTS
      IZ = IZ+1
      ANG = ATAN2(X(J),X(I))
C  THIS STATEMENT CHANGES THE ATAN2 RETURNED ANGLE FROM THE INTERVAL
C  -PI TO +PI TO THE INTERVAL 0 TO 2PI.
      IF (ANG.GT.(-PI).AND.ANG.LT. 0. )     ANG = ANG + TWOPI
      DO 10 IT=1,NSPD
C THE FOLLOWING IS NECESSARY TO MAKE ANGLES NEAR 2PI SEEM CLOSE TO
C   ANGLES NEAR 0 .
      IF (ANG.LT.  PI    ) GO TO 15
      IF (ABS(ANG-TWOPI-THET(IT)).LE.ANGTOL) GO TO 17
   15 IF(ABS(ANG-THET(IT)).GT.ANGTOL) GO TO 10
   17 DELTA = ABS((X(J)-YC)*COST(IT)-(X(I)-XC)*SINT(IT))
      DMAX = DELTA+DELXDM(IT)
      DMIN = DELTA-DELXDM(IT)
      IF(DMIN.GE.WDTHM) GO TO 10
      PER = 0.0
      IF(DMAX.LE.WDTHM) GO TO 20
      PER = SQRT((DMAX-WDTHM)/(DMAX-DMIN))
   20 CU(IZ) = CU(IZ)*PER
   10 CONTINUE
      RETURN
      END
```

32 SUBROUTINE SPTAN

The SPTAN subroutine shown in Figure 66 functions to take input values of x and y and return the angle whose tangent they represent. SPTAN insures that the angle returned is within the range

$$0 \le \Theta \le 2\pi$$

FUNCTION SPTAN 76/176 OPT=1 FIN 4.6+452 04/27/79 12.23.47

```
      FUNCTION SPTAN(X,Y)
      PI=3.14159265+
      SPTAN=0.0
      IF(X) 10,20,30
   10 SPTAN=PI+ATAN(Y/X)
      RETURN
   20 IF(Y) 21,22,23
   21 SPTAN=1.5*PI
   22 RETURN
   23 SPTAN=0.5*PI
      RETURN
   30 SPTAN=ATAN(Y/X)
      IF(Y.LT.0.0) SPTAN=SPTAN+2.*PI
      RETURN
      END
```

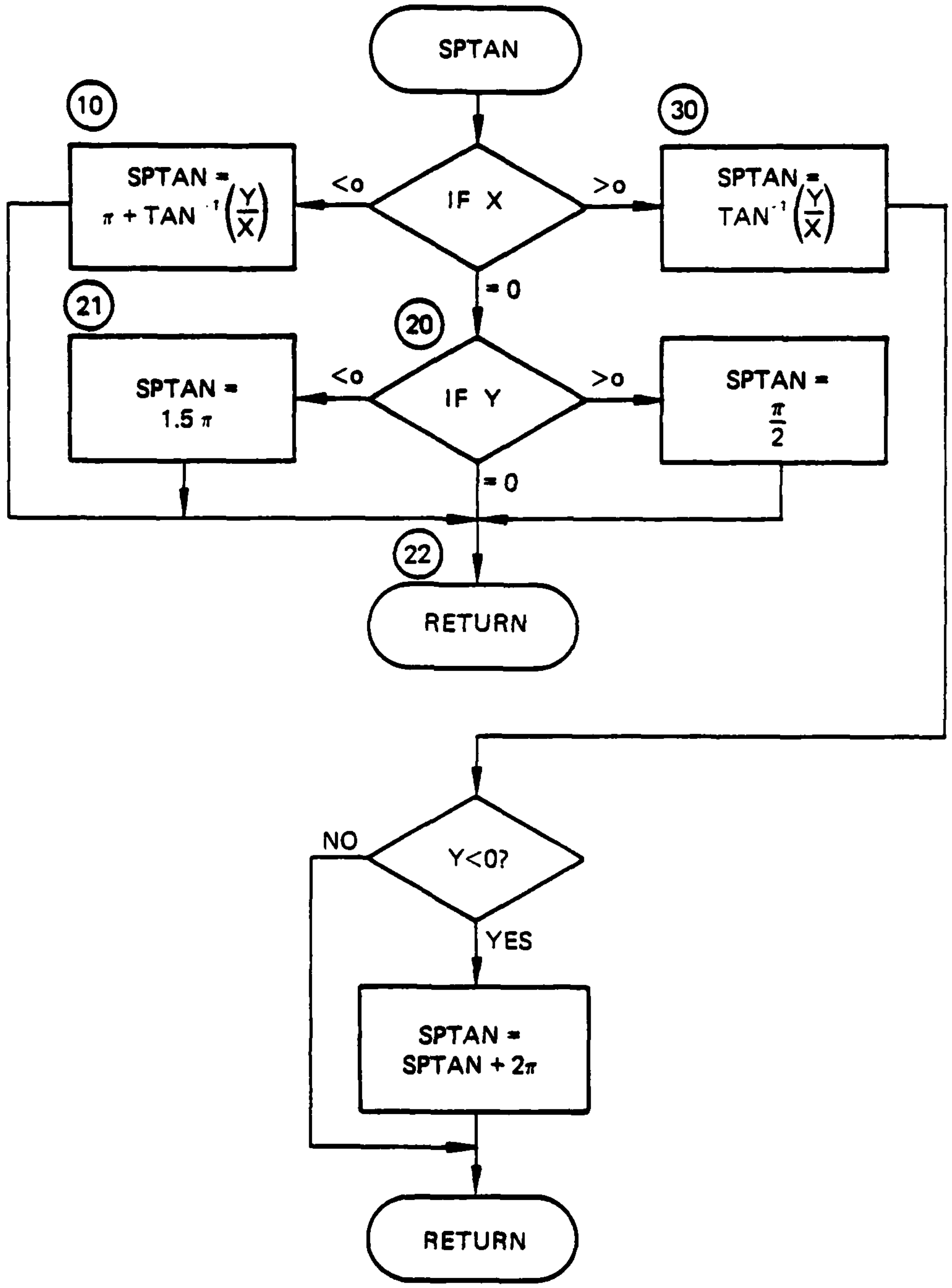


Figure 66. Subroutine SPTAN flow chart.

33. SUBROUTINE STEP

a. Purpose -- Subroutine STEP shown in Figure 67 is used to propagate the field through a vacuum. It also calculates Strehl intensity.

b. Relevant formalism

(1) Propagation -- STEP allows for two types of propagation

(a) Constant area mesh -- This type is used to propagate collimated and quasi-collimated beams. It assumes that edge spreading of the beam due to diffraction is not severe enough for the beam to get too close to the edge of the calculation region.

(b) Variable area mesh (VAMP) -- VAMP is used to propagate beams containing phase with curvature. As will be shown, the curvature is first removed from the field. The (collimated) field is then propagated an equivalent propagation distance which is defined by the formalism. After propagation, the propagated curvature is returned to the field.

The theory of VAMP propagation is developed in Section 5-D of AWFL-TR-73-231 and is repeated here for continuity.

First, consider constant area mesh propagation. The scalar wave function propagating in the Z-direction is written

$$\psi(\vec{x},t) = U(\vec{x})\varepsilon^{i(wt-kz)} \tag{198}$$

The function $\psi(x,t)$ obeys the scalar wave equation derived from Maxwell's equations

$$\nabla^2\psi = \frac{1}{c^2}\frac{\partial^2\psi}{\partial t^2} \tag{199}$$

If one assumes that

$$\frac{\partial^2\psi}{\partial t^2} << k\frac{\partial u}{\partial z} \tag{200}$$

then u(x) obeys the paraxial wave equation

$$\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} - 2ik\frac{\partial u}{\partial z} = 0 \tag{201}$$

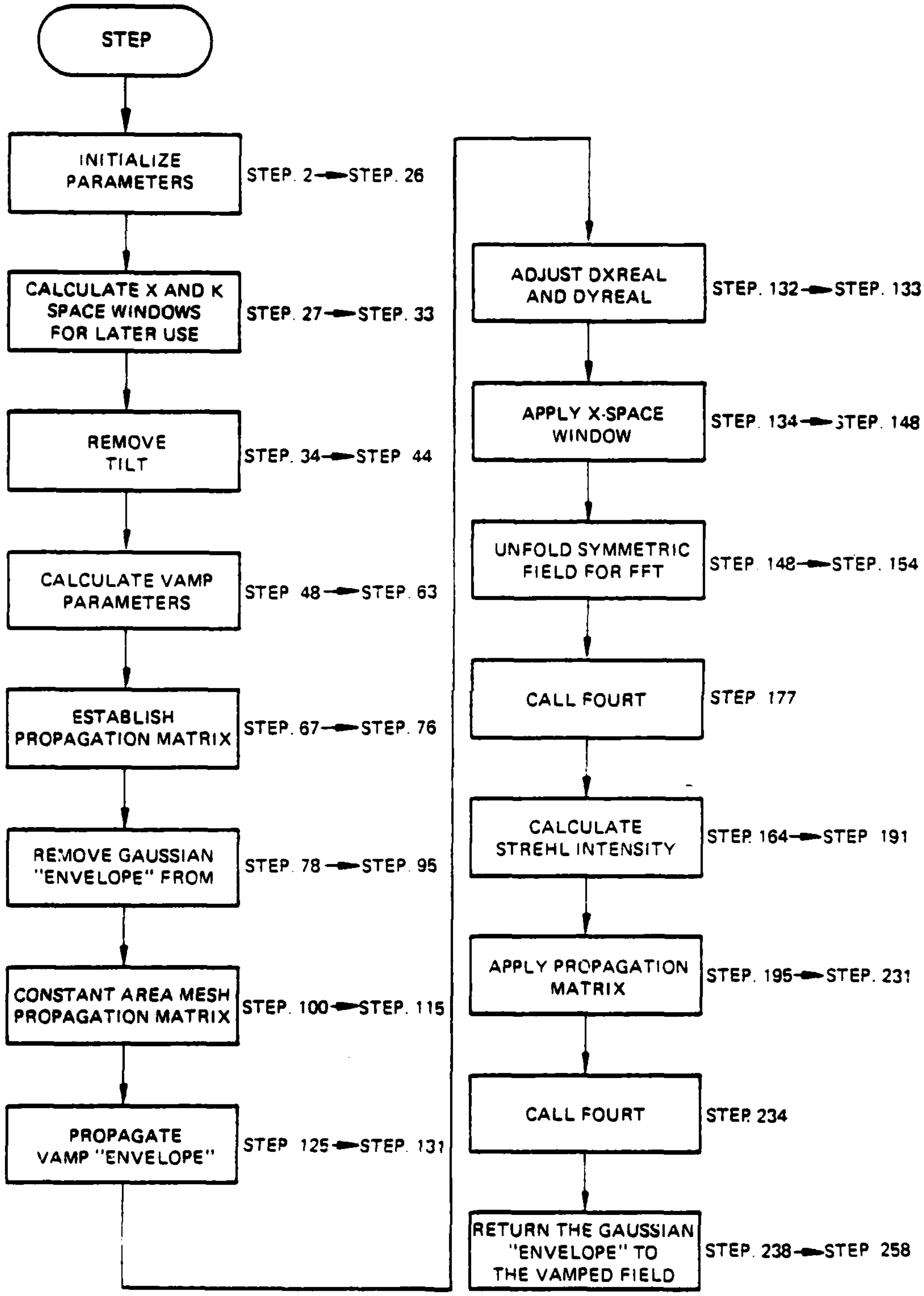


Figure 67. Subroutine STEP organization.

By using the method of Fourier Transforms u(x) is

$$u(\vec{x}) = \iint_{-\infty}^{\infty} df_x df_y \, \varepsilon^{2\pi i(f_x x + f_y y)} U(f_x x + f_y y) \varepsilon^{i\pi\lambda Z(f_x{}^2 + f_y{}^2)} \tag{202}$$

where

$$U(f_x, f_y) = \iint_{-\infty}^{\infty} dx'dy' \varepsilon^{-2\pi i(f_x x' + f_y y')} U(x', y', 0)$$

The Fourier Transforms are efficiently performed by using the FFT.

For variable area mesh, the following approach is used:

The spreading of the beam is estimated by that of a Gaussian reference beam with the same radius of curvature as the physical beam. This curvature is removed so that during propagation the beam continues to fill the calculation region.

Propagation of a Gaussian beam is easily handled by assuming knowledge of the associated Gaussian plane wave. According to Siegman, Chapter 8, (Ref. 14), a Gaussian plane wave (at Z = 0)

$$U_o(x_o, y_o) = \sqrt{\frac{2}{\pi}} \left(\frac{1}{w_o}\right) \varepsilon^{-(x_o^2 + y_o^2)/W_o^2} \tag{203}$$

when propagated a distance Z becomes

$$u(x,y,z) = \sqrt{\frac{2}{\pi}} \left(\frac{1}{w(z)}\right) \varepsilon^{-i(kz-\psi(z))} \varepsilon^{-(x^2+y^2)\left(\frac{k}{2R(z)} + \frac{1}{w(z)^2}\right)} \tag{204}$$

where

$$R(z) = z + \frac{z_R^2}{z} \qquad \psi(z) = \tan^{-1}\left(\frac{z}{z_R}\right)$$

$$w(z) = w_o \sqrt{1 + \left(\frac{z}{z_R}\right)^2}$$

14. Siegman, A. E., An Introduction to Lasers and Masers, McGraw-Hill, New York, 1971.

with

$$z_R = \frac{\pi w_o^2}{\lambda}\text{, the Rayleigh range.}$$

Therefore, to propagate a Gaussian beam of waist w(Z) and radius or curvature R(Z) a distance ΔZ, the following approach should be taken:

Knowing the waist and radius of curvature, one can determine the spot size w_o and distance to the spot size Z, according to

$$z_1 = \frac{R(z_1)}{1 + \left(\frac{\lambda R(z_1)}{\pi w(z_1)}\right)^2} \tag{205}$$

$$w_o = \frac{w(z_1)}{\sqrt{1 + \left(\frac{\pi w(z_1)^2}{\lambda R(z_1)}\right)^2}} \tag{206}$$

Then, from this origin a distance $Z_2 = Z_1 + \Delta Z$ is propagated to determine the desired wave function.

Since it is known how a Gaussian wave propagates, it is possible that transforming a given wave with a spherical wave front to Gaussian coordinates could result in the propagation of a quasi-collimated wave. The appropriate transformation is found to be

$$U(\bar{X}) = \frac{V(\bar{X})}{w(z)}\, \epsilon^{i\left[\frac{k(x^2 + y^2)}{2R(z)} + \tan^{-1}\left(\frac{z}{z_R}\right)\right]} \tag{207}$$

where Z is the distance from the current reference Gaussian beam, defined by R(Z) and w(Z) to its spot. Z_R is the Rayleigh range of this reference Gaussian beam.

By transforming to Gaussian coordinates:

$$X = x/w(z) \qquad Z = \tan^{-1}\left(\frac{z}{z_R}\right) \qquad Y = y/w(z) \tag{208}$$

The beam transformation is written as

$$u(\vec{x}) = v(\vec{x}) \frac{\cos Z}{w_o} \varepsilon^{-i(X^2+Y^2) \tan Z + iZ} \tag{209}$$

Inserting this equation into the paraxial wave equation results in the following differential equation in terms of Gaussian coordinates

$$-4i \frac{\partial v}{\partial Z} + \frac{\partial^2 v}{\partial X^2} + \frac{\partial^2 v}{\partial Y^2} + 4 \left(1-(X^2+Y^2)\right) v = 0 \tag{210}$$

which, except for the quadratic, is similar to the paraxial wave equation. The quadratic term $(X^2 + Y^2)v$ can be dropped if the reference Gaussian parameters and propagation distance are chosen so that v is equal to zero whenever X or Y approaches 1. This implies that the initial waist of the reference Gaussian be much larger than the size of the beam to be propagated. The propagation distance ΔZ must then be restricted so that the waist of the reference beam remains large compared with the beam size throughout the propagation. With these restrictions, the equation for v in Gaussian coordinates becomes

$$\frac{\partial^2 v}{\partial X^2} + \frac{\partial^2 v}{\partial Y^2} + 4v - 4i \frac{\partial v}{\partial Z} = 0 \tag{211}$$

As is the collimated case, Fourier Transform analysis gives the following result:

$$v(X,Y,Z) = \iint_{-\infty}^{\infty} df_x df_y \; V(f_x,f_y,Z) \varepsilon^{2\pi i(f_x X-f_y Y)} \tag{212}$$

where

$$V(f_x,f_y,Z) = V(f_x,f_y,Z)\varepsilon^{-i \left[1-\pi^2(f_x{}^2 + f_y{}^2)\right] (Z-Z_1)}$$

and

$$V(f_x, f_y, z_1) = \iint_{-\infty}^{\infty} dX dY \; V(X,Y,Z) \varepsilon^{-2\pi(f_x X + f_y Y)}$$

the propagated wavefunction is then v(X,Y,Z) multiplied by the propagation envelope:

$$u(x,y,z) = V(X,Y,Z) \frac{\cos Z}{w_o} \varepsilon^{i(X^2+Y^2) \tan Z + iZ} \tag{213}$$

where

$$X = \frac{x}{w(z)} \qquad Y = \frac{y}{w(z)} \qquad z_1 = \tan^{-1}\left(\frac{z}{z_x}\right)$$

z being the final distance from the reference spot. If the propagation takes place well outside of the Rayleigh range, Z is much greater than Z_R and the expansion of the arctangent for large argument can be used:

$$\begin{aligned} Z - Z_1 &= \tan^{-1}\left(\frac{z}{z_R}\right) - \tan^{-1}\left(\frac{z_1}{z_R}\right) \\ &= \left(\frac{\pi}{2} - \frac{z_R}{z}\right) - \left(\frac{\pi}{2} - \frac{z_R}{z_1}\right) \\ &= z_R \left(\frac{1}{z_1} - \frac{1}{z}\right) \end{aligned} \tag{214}$$

(2) Strehl intensity -- Since subroutine STEP propagates the beam using Fourier Transforms, the Strehl intensity is easily calculated.

The Strehl intensity gives an irradiation of the amount of aberration present in the beam at a given limiting aperture. It is defined as follows: Consider a field U(x,y). The field in the Frauenhofer diffraction region (the far field) is given by equations (4) through (13) in Goodman:

$$u(\vec{x}) = \varepsilon^{ikz} \varepsilon^{i\frac{k}{2z}(x^2+y^2)} \iint_{-\infty}^{\infty} u(\vec{x}') \varepsilon^{-\frac{2\pi i}{\lambda z}(\vec{x}.\vec{x}')} d\vec{x}' \tag{215}$$

Aside from the phase factor in front, this is just the Fourier Transform of the apertured field evaluated at

$$\vec{f} = \frac{\vec{x}}{\lambda z} \quad . \tag{216}$$

The Strehl intensity is defined as the ratio of the centerline intensity of the far field to that of a plane wave propagated the same distance coming from the same aperture with the same power. Analytically this is given as

$$I_{STREHL} = \frac{I_{CL-FF}}{I_{LL-PW-FF}} = \frac{\left| F\left(u(\vec{x}')\right) \right|^2_{\vec{f}=0}}{\left| F\left(u_{pw}(\vec{x}')\right) \right|^2_{\vec{f}=0}} \tag{217}$$

The plane wave centerline intensity is evaluated from

$$F\left(u_{pw}\ (\vec{x}')\right) = A_o \int_o^a r dr \int_o^{2\pi} d\theta \varepsilon^{2\pi f_r \cos\theta} \Big]_{\vec{f}=0} \tag{218}$$

$$= \pi a^2 A_o$$

A_o being the plane wave amplitude and a the radius of the aperture. Assuming a calculation region size of the L x L with N x N = total number of points, the centerline intensity of the far field for the real beam is found from

$$\begin{aligned} F\left(u(\vec{x}')\right) &= \iint_{-\infty}^{\infty} \overline{dx}\ u\ (\vec{x})\epsilon^{2\pi i \vec{f}\cdot\vec{x}} \\ &= \int_o^L dx \int_o^L dy\ \ u(x)\epsilon^{2\pi i \vec{f}\cdot\vec{x}} \\ &\simeq \sum_{I=1}^{N} \left(\frac{L}{N}\right) \sum_{J=1}^{N} \left(\frac{L}{N}\right) U\ (I,\ J)\epsilon^{2\pi i\left(\frac{L}{N}\right)(If_x + Jf_y)} \end{aligned} \tag{219}$$

where

$$\Delta X = \Delta y = \frac{L}{N} \qquad \text{and} \qquad x = I^*\left(\frac{L}{N}\right) \quad y = J^*\left(\frac{L}{N}\right)$$

assume

$$fx = \frac{KB}{N} \qquad \text{and} \qquad f_y = \frac{MB}{N}$$

where B is twice the maximum frequency of the spectrum of u,

then

$$F\left(u(\vec{x}')\right) \simeq F(K,M) = \left(\frac{L}{N}\right)^2 \sum_{I=1}^{N} \sum_{J=1}^{N} U(I,J)\varepsilon^{\,2\pi i \left(\frac{LB}{N}\right)\left(\frac{KI}{N} + \frac{MJ}{N}\right)} \tag{220}$$

But from the theory of discrete Fourier Transforms LB = N, so

$$F(K,M) = \left(\frac{L}{N}\right)^2 \sum_{I=1}^{N} \sum_{J=1}^{N} U(I,J)\varepsilon^{\,2\pi i\,(KI + MJ)/N} \tag{221}$$

The whole sum is just the (K,M) output of the FFT routine, so

$$F(K,M) = \left(\frac{L}{N}\right)^2 F_{FFT}(K, M) \tag{222}$$

The FFT returns the DC value (centerline) at F_{FFT} (1,1) so the Strehl intensity is defined as

$$I_{STREHL} = \frac{\left(\frac{L}{N}\right)^4 \left|F_{FFT}(1,1)\right|^2}{\left(\pi a^2\right)^2 I_o} \tag{223}$$

where $I_o = A_o^{\,2}$ = plane wave intensity.

Note: If the beam is not limited by an exit aperture just before the Strehl calculations, it is possible to have I_{STREHL} greater than one.

c. Fortran

Argument List

DELZ	=	Distance to be
RADCY	=	radius of curvature or the phase front
WINDOX	=	x-space cosine data window for FFT
WINDOK	=	K-space cosine data window for FFT
IFG	=	Vamp control parameter
	=	1 constant mesh
	=	2 variable mesh
ITR	=	Vamp control parameter
	=	0 stay in vamp
	=	1 transform back to constant mesh space
IPS	=	Tilt and defocus removal flag
	=	0 no correction
	=	1 remove tilt
	=	2 find defocus radius of curvature
	=	3 1 + 2 together
AX, AY	=	total beam tilt kep track of for beam placement in the inertial coordinate system instead of the beam coordinate system
NWRT	$\neq 0$	Propagates a wave distance DELZ without altering the stored value of total Z. NWRT = 1. Suppresses Strehl intensity calculation as well. NWRT = 1 when STEP is called from QUAL.
IFLAG	$\neq 0$	Assumes VAMP and/or CAMP parameters are established. It tells the routine to continue the propagation based on previous calculations of waist and curvature.

Common Variables Altered:

CU - becomes the propagated field

CFIL - is altered if IPS $\neq$ 0 by a call to TILT

X - altered if in VAMP

DXREAL - moved to keep track of center of beam in inertial frame as
DYREAL the beam propagates

WNOW - VAMP parameter altered to keep track of the current spot size

NREG - Flag to tell whether:

= 0: Constant area mesh propagation

= 1: VAMP inside half the Rayleigh range

= 2: VAMP outside twice the Rayleigh range

Other routines called:

TILT

FOURT

Computer printouts for subroutine STEP follow.

SUBROUTINE STEP 76/176 OPT=1 FIN 4.6+452 04/27/79 12.23.47

```
      SUBROUTINE STEP(DELZ,RADC4R,WNDOX,WNDOK,IFG,ITR,IPS,AX,AY,NWRT,
     X   IFLAG )
C GENERAL PROPAGATING ALGORITHM
C THIS ROUTINE IS USED TO PROPAGATE THE COMPLEX FIELD A DISTANCE
C DELZ  - IFLAG=1 IS USED WHEN CONTINUING WITH SAME PROPAGATING MATRIX
      LEVEL 2, CU,CUR
      COMMON/WAY/WNOW,NREG,RAPTR
      COMMON/MELT/CU(16384),CFIL(16512),X(128),WL,NPTS,NPY,DXREAL,DYREAL
      DIMENSION NNO(2),APR(2,2610),FACTR(64),CUR(32768),CDUMR(2)
      DOUBLE PRECISION W0,ZRAL,Z1,RADCUR,WW,WF,TNZ2
      COMPLEX CU,CFIL,CDUM
      EQUIVALENCE (CU(1),CUR(1)) , (CDUM,CDUMR(1))
      DATA ZINIT /0.0/
      IF (IFLAG.NE.0) GO TO 2000
      PI=3.141592
      NP2P2=NPTS+2
      NP = NPTS/2
      NPP1=  NP  +1
      ANP2=1.0/FLOAT(NPTS)**2
      NNO(1) = NPTS
      NNO(2) = NPTS
      NAR=2*NPTS*NPTS
      NREG=0
      RADCUR=RADC4R
      DCALC1=X(NPTS)-X(1)+X(2)-X(1)
      IF(WNDOX.LE.0.0) GO TO 48
      NWNDOX = WNDOX*FLOAT(NPTS)
```

```
C     X-SPACE COSINE DATA WINDOW
      DO 211  I=1,NWNDOX
  211 FACTR(I) = (1.0-COS(PI*FLOAT(I)/FLOAT(NWNDOX)))/2.0
   48 NWNDOK = WNDOK*FLOAT(NPTS)
      NO=NPP1-1-NWNDOK
      IF (IPS.NE.0) GO TO 1137
      IF (IFG.LT.1) GO TO 1137
      IF (IFG.GT.2) GO TO 1137
      IF (IFG.EQ.1) GO TO 1002
      GO TO 5
C     DETERMINE LINEAR AND QUADRATIC COMPONENTS OF PHASE
 1137 CALL TILT(AX,AY,RADCUR,IPS)
      IF (IFG.LT.1) GO TO 1139
      IF (IFG.GT.2) GO TO 1139
      IF (IFG.EQ.1) GO TO 1002
      GO TO 5
 1139 RWHEAK=1.E70
      IF (DABS(RADCUR/DELZ).GT.RWHEAK) GO TO 1002
C     ***********************************************************
C     VARIABLE AREA MESH PROPAGATION TRANSFORMATION TO EQUIVALENT
C     COLLIMATED BEAM
    5 ALPHA=10.
C     DETERMINATION OF BEAM WAIST AND DISTANCE TO IT
      W1 = ALPHA*DCALCI/2.
      WW = (W1*W1*PI/WL)**2
      Z1 = RADCUR*WW/(RADCUR**2+WW)
      W0 =DSQRT(DSQRT(RADCUR*Z1-Z1**2)*WL/PI)
      ZRAL = PI*W0*W0/WL
      ANZ=2.
      IF (DABS(Z1).LT.ZRAL/ANZ) NREG=1
      IF (DABS(Z1).GT.ZRAL*ANZ) NREG=2
      IF (NREG.EQ.0) GO TO 12
      IF (DABS(Z1+DELZ).GT.ZRAL/ANZ.AND.NREG.EQ.1) GO TO 12
      IF (DABS(Z1+DELZ).LT.ZRAL*ANZ.AND.NREG.EQ.2) GO TO 12
      DUME = PI**2*ZRAL/(DCALCI/W1)**2
      IPNT = 1
C     ESTABLISH PROPAGATING MATRIX
C     INCLUDES FREQUENCY SPACE DATA WINDOW
      DO 101  J=2,NPP1
      AJMISQ = (J-1)**2
      WFACTR = 1.0
      IF(J.GT.NO .AND. NWNDOK.GT.0)
     1 WFACTR = (1.0-COS(PI*FLOAT(NPP1-J)/FLOAT(NWNDOK)))/2.0
      DO 101 I=1,J
      DUM =(AJMISQ+(I-1)**2)
      IPNT = IPNT+1
      APR(1,IPNT)=WFACTR
  101 APR(2,IPNT)=DUME*DUM
      TNZ1 = Z1/ZRAL
      IJI=0
      DO 2 K=1,NPY
      YSQ = X(K)**2
      DO 2 I=1,NPTS
      IJI = IJI + 1
      IJI2 = IJI * 2
      IJI2M1 = IJI2 - 1
      PHI    =  (X(I)**2 + YSQ)*TNZ1/W1**2
      SINP   = SIN(PHI)
      COSP   = COS(PHI)
      CURS   = CUR(IJI2M1)
      CUR(IJI2M1) = W1*( CURS*COSP - CUR(IJI2)*SINP )
    2 CUR(IJI2)    = W1*( CURS*SINP + CUR(IJI2)*COSP )
      IF(NWRT.NE.0)ZKEEP=ZZZ
      ZZZ = Z1
      ZINTE=0.
      W3=W1
      IF (IFG.EQ.0) ITR=1
      GO TO 2000
C     ***********************************************************
C     CONSTANT AREA MESH PROPAGATION
C     INCLUDES FREQUENCY SPACE DATA WINDOW
```

```
 1002 ACDUM1=2.*PI/WL
      DUM1 = (WL/DCALC1)**2
      IPNT = 1
C     ESTABLISH PROPAGATING MATRIX
      DO 200  J=2,NPP1
      AJM1SQ = (J-1)**2
      WFACTR = 1.0
      IF(J.GT.N0 .AND. NWNDOX.GT.0)
     1 WFACTR = (1.0-COS(PI*FLOAT(NPP1-J)/FLOAT(NWNDOX)))/2.0
      DO 200 I=1,J
      DUM =(AJM1SQ+(I-1)**2)
      DUM2 = DUM1*DUM
      DUM3 = (0.125*DUM2+0.5)*DUM2
      IPNT = IPNT+1
      APR(1,IPNT)=WFACTR
  200 APR(2,IPNT)=ACDUM1*DUM3
C     ENTER ROUTINE HERE WHEN CONTINUING WITH SAME PROPAGATING MATRIX
C     ENTRY CORE(DELZ,ITR,NWRT)
 2000 ZZZ=ZZZ+DELZ
      IF(NWRT.NE.0) GO TO 402
      ZINTE=ZINTE+DELZ
      ZZMIN=ZINTE-DELZ
      XMESH = X(NPTS)-2.*X(1)+X(2)
      RCEK=(1.-2.*WNDOX)*X(NPTS)
      IF(RAPTR.GE.RCEK)RAPTR=0.
  402 IF (NNEG.EQ.0) GO TO 92
      WNOW=W0*DSQRT(1.+(ZZZ/ZRAL)**2)
      XXPAND=WNOW/W3
      W3=WNOW
C     ADJUST BEAM COORDINATES FOR MAGNIFICATION AND MIRROR TILT
      DO 93 I=1,NPTS
   93 X(I)=X(I)*XXPAND
   92 DXREAL=DXREAL+ SIN (AX) * DELZ
      DYREAL=DYREAL+ SIN (AY) * DELZ
      IF(WNDOX.LE.0.0) GO TO 49
C     APPLY X-SPACE COSINE DATA WINDOW
      DO 212  I=1,NPTS
      DO 212  J=1,NWNDOX
      IJ2 = I + (NPTS     -J) * NPTS
      IF (NPY.EQ.NPTS) CU(IJ2) = CU(IJ2) * FACTR(J)
      IJI=I+(J-1)*NPTS
  212 CU(IJI)=CU(IJI)*FACTR(J)
      DO 213  J=1,NPY
      IJ = (J-1)*NPTS
      DO 213  I=1,NWNDOX
      I2=NPTS+1-I
      CU(I+IJ)=CU(I+IJ)*FACTR(I)
  213 CU(I2+IJ)=CU(I2+IJ)*FACTR(I)
C     UNFOLD SYMETRIC FIELD FOR FFT USE
   49 IF (NPTS.EQ.NPY) GO TO 50
      DO 15 J=1,NPY
      DO 15 I=1,NPTS
      IJ = I +NPTS*(J-1)
      IJI = I + (NPTS-J)*NPTS
   15 CU(IJI)= CU(IJ)
C   ****************    STREHL  INTENSITY CALCULATION  ****************
C   *   STREHL INTENSITY IS CALCULATED FROM THE CENTERLINE INTENSITY  *
C   *   OF THE FAR FIELD DISTRIBUTION. THE METHOD USES THE CENTERLINE *
C   *   COEFFICIENT OF THE FFT FOR THE UNNORMALIZED CENTERLINE        *
C   *   INTENSITY.  POWER CONSERVATION IS USED TO DEFINE THE PLANE WAV*
C   *   NEAR FIELD INTENSITY VALUE. THE RATIO OF CENTERLINE INTENSITY *
C   *   (FFT) TO PEAK INTENSITY (PLANE WAVE) DEFINES STREHL INTENSITY.*
C   *   IN THIS ROUTINE.  J FORGHAM 10 28 74                          *
C   ***************************************************************
   50 IF(RAPTR.EQ.0.0.OR.NWRT.EQ.1)GO TO 96
      XITOT  = 0.
      PI     = 3.141596
      XMESH4 = XMESH**4.
      NOB=NPTS*NPTS
      DO 95 I=1,NOB
      I2 = I * 2
```

```
      XITOT  =  XITOT  +  CUR(I2-1)**2 + CUR(I2)**2
   95 CONTINUE
C     XITOT = INTEGRAL OF INTENSITY (UNNORMALIZED)
C     CU(I) CONTAINS CENTER LINE FFT OF NEAR FIELD DISTRIBUTION AFTER
C     RETURN FROM " FOURT".
C     TRANSFORM COMPLEX FIELD TO FREQUENCY SPACE WITH FFT
   96 CALL FOURT(CU,NAR,NND,1)
      IF(RAPTH.EQ.0.0.OR.NWRT.EQ.1)GO TO 99
      AREA = PI*RAPTH**2
      AREASQ = AREA * AREA
      XIBAR = XITOT / NOB
      XIBRPW = XIBAR *((XMESH*XMESH)/AREA)
C *** XIBRPW = PLANE WAVE INTENSITY (NEAR FIELD)
      NOBSQ  = NOB * NOB
      XINORM = XMESH4 / NOBSQ
      CLIFF = (CUR(1)**2 + CUR(2)**2) * XINORM
C     CLIFF = CENTERLINE INTENSITY (FAR FIELD)
C     STREHL INTENSITY
      STRINT = CLIFF / (XIBRPW* AREASQ)
      WRITE (6,16) STRINT
   16 FORMAT(///2X,19H STREHL INTENSITY =       ,G12.5)
   99 RAPTH=0.0
      DTZ=DELZ
C     CALCULATE DELZ IN EQUIVALENT COLLIMATED COORDINATE SYSTEM
      IF (NREG.EQ.1) DTZ=DATAN(ZZZ/ZRAL)-DATAN((ZZZ-DELZ)/ZRAL)
      IF (NREG.EQ.2) DTZ=DELZ/(ZZZ*(ZZZ-DELZ))
      IPNT = 1
      CU( 1 )=CU( 1 )*ANP2
C     APPLY PROPAGATION MATRIX
      DO 100 J=2,NPP1
      J1 = NP2P2-J
      DO 100 I=1,J
      I1 = NP2P2-I
      IPNT = IPNT+1
      PHI  = DTZ * APR(2,IPNT)
      SINP = SIN(PHI)
      COSP = COS(PHI)
      ACNST = ANP2 * APR(1,IPNT)
      CDUMR(1) = ACNST * COSP
      CDUMR(2) = ACNST * SINP
C     CDUM=ANP2*APR(1,IPNT)*CEXP(CMPLX(0.,APR(2,IPNT)*DTZ))
      CU(I+NPTS*(J-1)) = CU(I+NPTS*(J-1))*CDUM
      IF(I.EQ.J) GO TO 108
      CU(J+NPTS*(I-1)) = CU(J+NPTS*(I-1))*CDUM
      IF(J.EQ.NPP1) GO TO 109
      CU(I+NPTS*(J1-1)) = CU(I+NPTS*(J1-1))*CDUM
      CU(J1+NPTS*(I-1))= CU(J1+NPTS*(I-1))*CDUM
      IF(I.LT.2) GO TO 100
       CU(I1+NPTS*(J-1)) = CU(I1+NPTS*(J-1))*CDUM
      CU(J+NPTS*(I1-1)) = CU(J+NPTS*(I1-1))*CDUM
      CU(I1+NPTS*(J1-1)) = CU(I1+NPTS*(J1-1))*CDUM
      CU(J1+NPTS*(I1-1)) = CU(J1+NPTS*(I1-1))*CDUM
      GO TO 100
  108 IF(I.EQ.NPP1) GO TO 100
      CU(I+NPTS*(J1-1)) = CU(I+NPTS*(J1-1))*CDUM
      CU(J1+NPTS*(I-1)) = CU(J1+NPTS*(I-1))*CDUM
      CU(I1+NPTS*(J1-1)) = CU(I1+NPTS*(J1-1))*CDUM
      GO TO 100
  109 IF(I.LT.2) GO TO 100
      CU(I1+NPTS*(J-1)) = CU(I1+NPTS*(J-1))*CDUM
      CU(J+NPTS*(I1-1)) = CU(J+NPTS*(I1-1))*CDUM
  100 CONTINUE
C     TRANSFORM COMPLEX FIELD TO X-SPACE WITH FFT
      CALL FOURT(CU,NAR,NND,-1)
      IF (NWRT.NE.0) ZZZ=ZKEEP
      IF (ITH.EQ.0.OR.NREG.EQ.0) RETURN
C     TRANSFORM FROM EQUIVALENT COLLIMATED COORDINATE SYSTEM (X¢,Y¢)
C     BACK TO REAL COORDINATE SYSTEM (X,Y).
      WF = W0*DSQRT(1.+(ZZZ/ZRAL)**2)
      TNZ2 = ZZZ/ZRAL
      FF=TNZ2/(WF*WF)
      DO 42 J=1,NPY
```

```
      YSQ = X(J)**2                                                     STEP
      DO 42 I=1,NPTS                                                    STEP
      IJ1 = I+(J-1)*NPTS                                                STEP
      IJ12 = 2 * IJ1                                                    STEP
      IJ12M1 = IJ12 - 1                                                 STEP
      PHI = -(X(I)**2 + YSQ) *FF                                        STEP
      SINP = SIN (PHI)                                                  STEP
      COSP = COS(PHI)                                                   STEP
      CURS = CUR(IJ12M1)                                                STEP
      CUR(IJ12M1) = (CURS*COSP - CUR(IJ12)*SINP)/WF                     STEP
   42 CUR(IJ12)   = (CURS*SINP + CUR(IJ12)*COSP)/WF                     STEP
      XXPAND=WF/W1                                                      STEP
      NREG = 0                                                          STEP
      WRITE (6,522) XXPAND                                              STEP
  522 FORMAT (/37H   THE MAGNIFICATION OF THE FIELD IS ,F10.6/)         STEP
      RETURN                                                            STEP
   12 WRITE (6,9)                                                       STEP
    9 FORMAT(///,33H    INVALID VARIABLE MESH REGION ,/,53H     SUBROUTI STEP
     1NE STEP COUNTINUING WITH CONSTANT MESH  ,/,65H   NOTE POSSIBLE EXP STEP
     1ANSION OF THE BEAM OUTSIDE THE CALC. REGION  ,///)                STEP
      IFG=1                                                             STEP
      NREG=0                                                            STEP
      GO TO 1002                                                        STEP
      END                                                               STEP
```

34. SUBROUTINE TBLOOM

a. Purpose -- This subroutine, shown in Figure 68, is used to model four types of thermal blooming which may be seen by a beam as it propagates through an absorptive medium.

The four types are:

1. Tranverse
2. Axial
3. Free convective
4. Transient

b. Relevant formalism -- Thermal blooming arises as a consequence of the absorption of laser radiation by the transmitting gas. The absorbed radiation heats the gas and consequently changes its refractive index. These variations in the index of refraction induce phase changes in the propagated beam. Phase changes produced by thermal blooming can result in beam divergence, which overloads apertures and provides a source of high energy feedback. Thermal blooming also degrades beam quality. Thermal blooming models are available in the SOQ library to describe the impact on the beam phase and amplitude produced when thermal blooming occurs in (1) a transverse flow field, (2) an axial flow field, (3) a free convective flow field, and (4) transient conditions with no external flow.

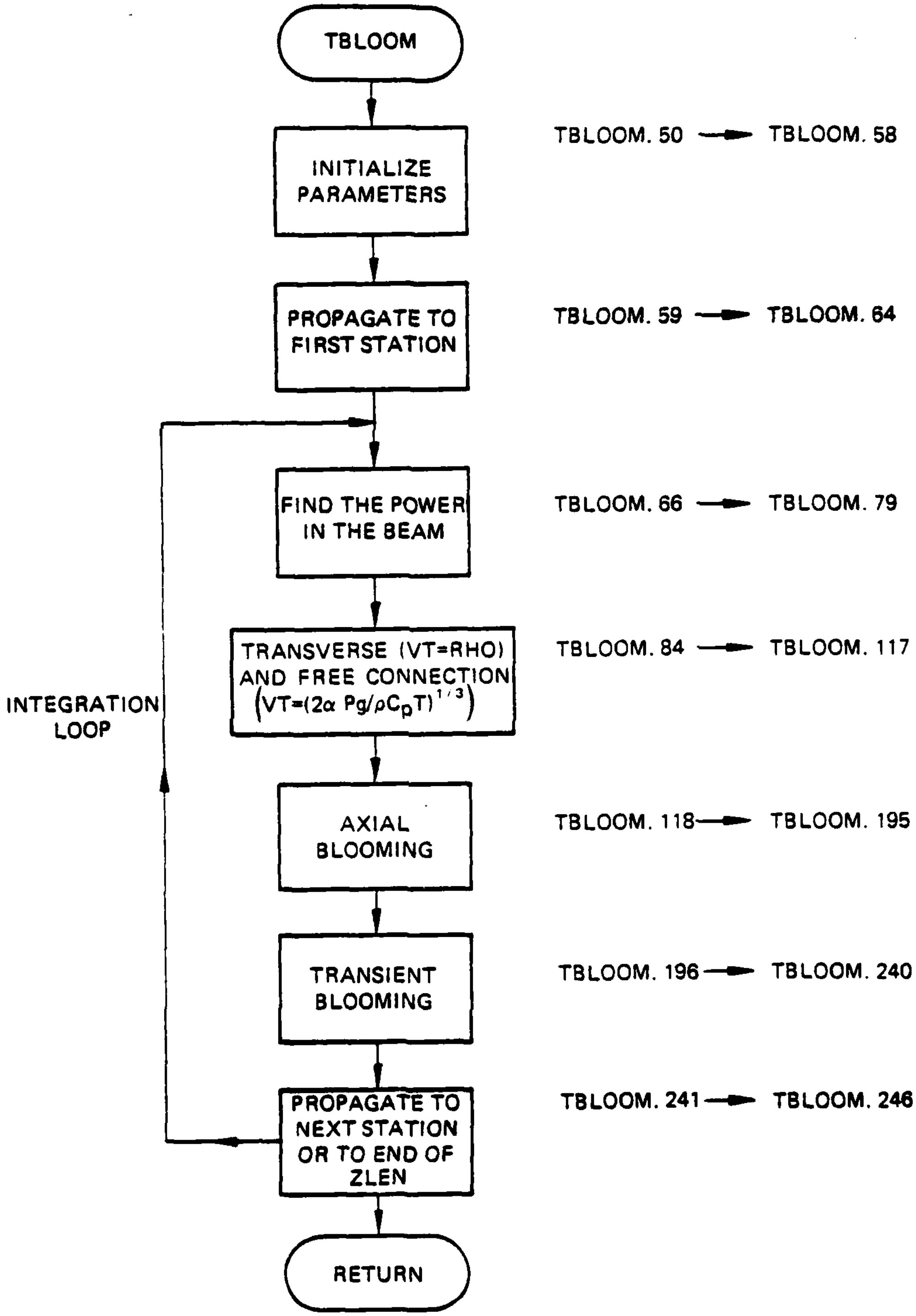


Figure 68. Subroutine TBLOOM flow chart.

Figure 69 schematically demonstrates the procedure used to modify the complex field, U(x,y), as it is propagated through a thermal blooming gain phase segment within the SOQ code.

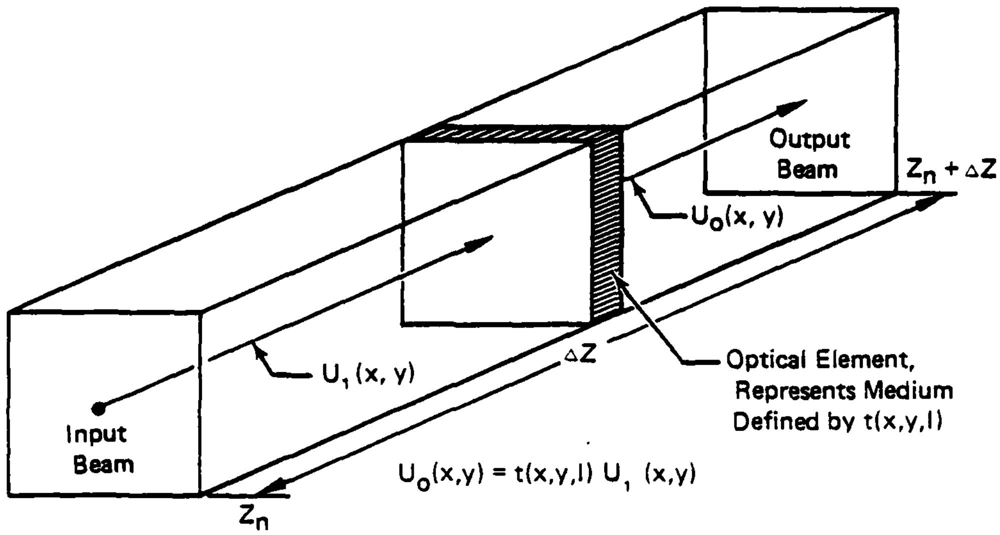


Figure 69. Illustration of thermal blooming model.

As the beam is propagated a distance ΔL through the medium, it is continuously interacting with that medium. By requiring that the effect is small, the integrated effect can be approximated by a finite number of discrete steps in the following manner:

Assume each step is of length ΔL and that the effect of such a step is approximated by a vacuum propagation to the center (ΔL/2), application of the appropriate transmission function t(x,y,I), followed by subsequent vacuum propagation of field the remaining distance (ΔL/2).

The transmission function t(x,y,ΔL,I(x,y) can be assumed to be of the form

$$t(x,y,I) = \exp\left[-\frac{\alpha\Delta L}{2} - i\Delta\phi\right] \tag{224}$$

where α is the absorptivity of the medium and Δϕ can be written

$$\Delta\phi = \frac{2\pi}{\lambda}\frac{dn}{dt}\int_0^{\Delta L} \delta T \, dz' \tag{225}$$

$$\delta T = \delta T\,(x,y,z')$$

Employing the usual Gladstone-Dale relationship to approximate the index n, ($n = 1+\rho C$) and the equation of state for an ideal gas ($P = \frac{RT}{M}\rho$), the expression for $\Delta\phi$ becomes (assuming constant pressure)

$$\Delta\phi = \frac{2\pi}{\lambda}\left(-\frac{\rho C}{T}\right)\int_0^{\Delta L} dz\ \delta T\,(x,y,z) \tag{226}$$

δT represents the temperature variation across the beam as a result of one of the four types of thermal blooming. It is found in the following manner:

(1) Transverse blooming -- It is assumed that the wind is blowing with speed V_T (con/scan) from the negative x-direction. The resulting temperature variation is:

$$\delta T_T = \frac{\alpha}{\rho C_p\, V_T}\int_{-\infty}^{x} I\,(x',y,z)\, dx' \tag{227}$$

where I is the intensity of the beam.

(2) Axial blooming -- It is assumed that the wind blows in the same direction the beam is traveling with speed V (cm/sec) resulting in

$$\delta T_{ax} = \frac{\alpha}{\rho C_p\, V_{ax}}\int_0^{x} I\,(x,y,z')\,dz' \tag{228}$$

(3) Free convection -- The temperature variation due to thermal gradients caused by absorption is:

$$\delta T_c = \frac{\alpha}{\rho C_p\, V_c}\int_{-\infty}^{x} I\,(x',y,z)\,dx' \tag{229}$$

where $$V_c = \left(\frac{2\alpha P z g}{\rho C_p T}\right)^{\frac{1}{3}}$$

P(Z') being the total power in the beam at Z' and g, the acceleration due to gravity.

(4) Transient -- Finally, in the process of establishing free convection, the beam has a residence time $T_{(sec)}$ during which the temperature variation is

$$\delta T_{tran} = \frac{\alpha\tau}{\rho C_p} + I \qquad (230)$$

c. Fortran

Argument List

ALFA - Absorptivity of the medium (cm^{-2})

CP - Specific heat (J/g-K)

T - Temperature (K)

RHO - (1) if RHO$\leq$1, it is the density (g/cm^3) used for free convection

(2) if RHO$\geq$1, it is the transverse velocity

ZLEN - Total length of the blooming medium

NSTEPS - The number of steps required to adequately represent thermal blooming over a distance ZLEN. Phase per step shift usually kept $\leq \frac{\pi}{8}$ /8

INPT - Flag for intermediate plots

NPROP - Same as NSTE in cavity

AXIAL - Axial velocity (cm/sec) and is > 0

DT - Residency time for transient blooming

None of the above parameters is redefined by this subroutine.

Commons:

The variables in common which are modified are:

(1) CU: the effect of the blooming is applied to CU

(2) CFIL: due to its equivalence with the PH and W arrays, it is modified when they are defined.

Computer printouts of subroutine TBLOOM follow.

SUBROUTINE TBLOOM   76/176   OPT=1   FIN 4.6+452   04/27/79   12.23.47

```
      SUBROUTINE TBLOOM(ALFA,CP,T,RHO,ZLEN,NSTEPS,INPT,NPROP,AXIAL,DT)   TBLOOM
      LEVEL 2, CU,CUR,W,PH                                               TBLOOM
      COMMON/MELT/CU(16384),CFIL(16512),X(128),WL,NPTS,NPY,DRX,DRY       TBLOOM
      COMMON/WAY/WNO,,NREG,RAPTH                                         TBLOOM
      DIMENSION W(16384),PH(16384)                                       TBLOOM
      REAL CUR(32768)                                                    TBLOOM
      REAL ISAT                                                          TBLOOM
      COMPLEX CU,CFIL                                                    TBLOOM
      EQUIVALENCE (CU(1) , CUR(1))                                       TBLOOM
C     *******************          *********   ********************      TBLOOM
C           THIS  VERSION  OF  TBLOOM  HAS  BEEN  MODIFIED  TO           TBLOOM
C           ACCOMODATE AXIAL  BLOOMING CALCULATIONS  PER PHASE           TBLOOM
C           TWO-THREE  PROPOSAL  J FORGHAM  6/ 75                        TBLOOM
C     *******************          *********   ********************      TBLOOM
C   THIS ROUTINE HAS BEEN FURTHER MODIFIED TO ACCOMODATE TRANSIENT       TBLOOM
C   THERMAL BLOOMING CALCULATIONS. TRANSIENT TH.BL. IS THE PHASE         TBLOOM
C   CHANGE  WHICH RESULTS FROM ENERGY ADDITION TO THE MEDIUM             TBLOOM
C   WITH NO FORCED OR FREE CONVECTION . WE SOLVE.......                  TBLOOM
C         RHO * CP * DTEMP/DTIME = ALPHA * I(X,Y,Z)                      TBLOOM
C   AND FIND PHASE CHANGE FROM THE LINEARIZED INDEX CHANGE...            TBLOOM
C         DELTA N = DN/DTEMP *DELTA TEMP                                 TBLOOM
C                FORGHAM  12 / 19 /74                                    TBLOOM
C                                                                        TBLOOM
C     *******************          ****  *****   ******************      TBLOOM
      EQUIVALENCE (W( 1 ),CFIL( 1 )),(PH( 1 ),CFIL(8193))                TBLOOM
      NST=NPROP                                                          TBLOOM
      M  = 0                                                             TBLOOM
      IOUT = 1                                                           TBLOOM
      IF (NPROP.EQ.3.OR.NPROP.EQ.5) IOUT = 0                             TBLOOM
      IF (NPROP.EQ.3) NST=2                                              TBLOOM
      WRITE(6,5) ALFA,CP,T,     ZLEN,NSTEPS                              TBLOOM
    5 FORMAT(119H0FIELD HAS ENTERED SUBSYSTEM TBLOOM - STEADY STATE THER TBLOOM
     XMAL BLOOMING MEDIUM                                             /2 TBLOOM
     X5X,25HABSORPTION COEFFICIENT = ,G12.5,5H CM-1/25X,                 TBLOOM
     X19HSPECIFIC HEAT,CP = ,G12.5,7H J/GM-K/25X,                        TBLOOM
     X14HTEMPERATURE = ,G12.5,7H DEG. K/25X,                             TBLOOM
     X12HTHICKNESS = ,G12.5,3H CM/25X,                                   TBLOOM
     X15HNO. ELEMENTS = ,I3)                                             TBLOOM
      IF(DT.GT.0.0) GO TO 700                                            TBLOOM
C     ******** DT GREATER THAN 0.0 INDICATES TRANSIENT BLUMING *********  TBLOOM
      IF(AXIL .GT.0.) WRITE(6,596)AXIAL                                  TBLOOM
  596 FORMAT(25X,18HAXIAL VELOCITY =  ,G12.5, 8H CM/SEC )                TBLOOM
      IF(AXIAL .GT. 0.) GO TO 700                                        TBLOOM
C     ******** AXIAL = AXIAL VELOCITY ****                               TBLOOM
      IF (RHO .LT. 1.) WRITE(6,6) RHO                                    TBLOOM
    6 FORMAT(25X,10HDENSITY = ,G12.5,7H GM/CM3)                          TBLOOM
      IF (RHO .GT. 1.) WRITE(6,7) RHO                                    TBLOOM
    7 FORMAT(25X,23HTRANSVERSE VELOCITY  = ,G12.5,7H CM/SEC)             TBLOOM
  700 DELZ = ZLEN/NSTEPS                                                 TBLOOM
      GDC = .223                                                         TBLOOM
      RAD = 1.                                                           TBLOOM
      ZLAST = 0.                                                         TBLOOM
      ZNOW  = 0.                                                         TBLOOM
      AVELAG  =  0.                                                      TBLOOM
      RMSTOT  =  0.                                                      TBLOOM
      PHTOT   =  0.                                                      TBLOOM
      PRED = EXP(-ALFA*DELZ/2.0)                                         TBLOOM
C *** PROPAGATE TO FIRST ELEMENT                                         TBLOOM
C      IF (  NPROP.GE.4  ) CALL CORE(DELZ/2.,0,M)                        TBLOOM
       IF (  NPROP.GE.4  )                                               TBLOOM
     1CALL STEP(DELZ/2.,RAD,.1,.1,NST, 0,0,0.,0.,M,1)                    TBLOOM
       IF (  NPROP.LE.3  )                                               TBLOOM
     1CALL STEP(DELZ/2.,RAD,.1,.1,NST, 0,0,0.,0.,M,0)                    TBLOOM
       DO 100 K=1,NSTEPS                                                 TBLOOM
       KM1=K-1                                                           TBLOOM
```

```
      DX = X(2) -X(1)
      DXSQ = DX**2
      DCAL = NPTS*DX
      XFACT = 1.
      IF(NREG.EQ.1.OR.NREG.EQ.2)XFACT = 1./WNOW**2
C *** COMPUTE POWER DENSITY
      NOB=NPTS*NPY
      PT = 0.
      DO 10 I=1,NOB
C     W( I ) = CU( I )*CONJG(CU( I ))*XFACT
      W( I ) = (CUR(2*I-1)**2 + CUR(2*I)**2) *XFACT
   10 PT = PT+W( I )
      PT = PT*DXSQ*NPTS/NPY
      IF(DT.GT.0.0) GO TO 220
C *** TEST DT TO DETERMINE IF TRANSIENT BLOOMING REQUIRED
      IF ( AXIAL  .GT. 0. ) GO  TO  18
C *** TEST AXIAL TO DETERMINE IF AXIAL BLOOMING IS REQUIRED
      VT = RHO
      IF(RHO .LT. 1.0)
     AVT = (980.665*PT*ALFA/(RHO*CP*T))**(1./3.)
      CAPK = 6.2831853*ALFA*DELZ*DCAL/(WL*CP*T*VT)*GDC
      IF(INPT.EQ.0) GO TO 15
      IF(MOD(KM1,INPT).NE.0)GO TO 15
      WRITE(6,14)K,PT,VT,CAPK
   14 FORMAT(46H1 FIELD INCIDENT UPON THERMAL BLOOMING ELEMENT,I2,8H POW
     1ER= ,G12.5,23H TRANSVERSE VELOCITY = ,G12.5,15HCM/S    CAPK = ,G12
     1.5)
      N  = 0
      UMAX  = 0.
      CALL OUTPUT(CU,NPY,NPTS,X,N,UMAX,.TRUE.,.FALSE.,.FALSE.)
   15 PMAX = -1.E7
      WAIST2 = 25
   19 CONTINUE
      DO 20 J=1,NPY
      SUM = 0.
      J1=(J-1)*NPTS
      DO 20 I=1,NPTS
      JJ=I+J1
      SUM = SUM+W( JJ)
      PH( JJ) = CAPK*SUM/NPTS
      CU( JJ) = CU( JJ)*CMPLX(COS(PH( JJ)),SIN(PH( JJ)))*PRED
   20 IF(PH( JJ).GT.PMAX)PMAX=PH( JJ)
      IF(INPT.EQ.0) GO TO 35
      IF(MOD(KM1,INPT).NE.0)GO TO 35
      WRITE(6,34) K,PMAX
   34 FORMAT(54H1 FIELD AFTER MODIFICATION BY THERMAL BLOOMING ELEMENT,I
     12,32H MAXIMUM PHASE SHIFT INDUCED WAS,G12.5,8H RADIANS)
      N  = 0
      UMAX  = 0.
       CALL OUTPUT(CU,NPY,NPTS,X,N,UMAX,.TRUE.,.FALSE.,.FALSE.)
       GO TO  35
C      *****************     **********          **************
C      THIS SECTION IS DESIGNED TO CALCULATE PHASE CHANGE OF THE BEAM
C      DUE TO AN AXIAL VELOCITY COMPONENT. THE MATH REQUIRES THE SOLEN
C      OF THE ENERGY EQUATION FOR A TEMP RISE PARALLEL TO THE BEAM AXIS.
C      IN WHAT FOLLOWS, CAPKAX IS A DISTORTION NUMBER  OF SORTS, AND
C      THE PHASE CHANGE AT EACH MESH POINT RESULTS FROM THE PRODUCT
C      OF CAPKAX * INTENSITY "W". THE FIELD IS MODIFIED BY THE PHASE
C      CHANGE INDUCED,AND THE POWER LOST TO HEATING THE MEDIUM "PRED".
C      *******************      ********          ****************
   18 CAPKAX  =       6.2831853*ALFA*GDC          /(WL*CP*AXIAL*T*2.)
      ZNOW  =   ZNOW  + DELZ
      IF (INPT .EQ. 0 ) GO TO 50
      IF(MOD(KM1,INPT).NE.0)GO TO 50
      WRITE(6,45)K,PT,AXIAL,CAPKAX
      WRITE (6,46) ZNOW
   45 FORMAT(46H1 FIELD INCIDENT UPON THERMAL BLOOMING ELEMENT,I2,8H POW
     1ER= ,G12.5,23H  AXIAL     VELOCITY = ,G12.5,15HCM/S  CAPKKAX= ,
     2 G12.5)
   46 FORMAT(10X,19HAXIAL  POSITION =  ,G12.5,3H CM)
      N  = 0
      UMAX  = 0.
```

```
      CALL OUTPUT(CU,NPY,NPTS,X,N,UMAX,.TRUE.,.FALSE.,.FALSE.)
C     ******** THE DO 200 LOOP IS AN ANALYTICAL  GAUSSIAN BLOOM *****
C     ******** THE DO 200 ALSO CALCULATES PHASE-GAIN NUMERICALY *****
   50 PMAXAX =-1.E+7
      EWAIST = 5.0
      PHBAR = 0.0
      PHSQ   = 0.0
      DO 200 J = 1,NPY
      J1=(J-1)*NPTS
      DO 200 I = 1,NPTS
C     ARG   = X(I) * X(I) + X(J)*X(J)
C     WAIST2 = EWAIST * EWAIST
C     IF (ARG     .GE. WAIST2 ) ARG = 0.0
C     PHGAUS = CAPKAX * (PT / 3.14159)*(1./WAIST2) *(EXP((-ARG * 2.)/
C    X  WAIST2)) * 2.31*(ZNOW**2- ZLAST**2)
      KK  = I + J1
      PH(KK) = CAPKAX * W(KK)*(ZNOW**2  -ZLAST**2)
      CU(KK) = CU(KK) * CMPLX(COS(PH(KK)),SIN(PH(KK))) * PRED
C     DELTA = PHGAUS  -  PH(KK)
      PHBAR = PHBAR  + PH(KK)
      PHSQ  =  PHSQ  +  PH(KK) * PH(KK)
C     IF (J .NE.  1 + NPY/2 ) GO TO 181
C     IF (INPT .EQ. 0 )GO TO 1798
C     WRITE (6,180)X(I),X(J),PHGAUS,PH(KK),DELTA
C1798 CONTINUE
C 180 FORMAT(5X,5G12.5)
C 181 CONTINUE
  200 IF (PH(KK) .GT. PMAXAX) PMAXAX = PH(KK)
C       ***************      *******      ****************************
C       RMSPHS = RMS PHASE DISTORTION FOR DELZ STEP
C       AVELAG = AVERAGE PHASE LAG FOR THERMAL BLOOMING SEGMENT
C       PHBAR1 = AVERAGE PHASE LAG FOR DELZ STEP
C       RMSTOT = TOTAL RMS PHASE FOR THERMAL BLOOMING SEGMENT
C       PHTOT  = TOTAL MAXIMUM PHASE LAG FOR THERMAL BLOOOMING SEGMENT
C       THE ABOVE SISTISTICAL PARAMETERS  ARE INCLUDED AS DIAGNOSTICS
C       **************** JLF 8/26/74  *****************************
C      RMSPHS  =  SQRT( PHSQ  -  ( ( PHBAR**2)/(NPY*NPTS)))
C      TOTPTS  = NPY * NPTS
C      RMSPHS  = RMSPHS / SQRT(TOTPTS  )
C      PHBAR1  = PHBAR /(NPY*NPTS)
C      AVELAG  = AVELAG + PHBAR1
C      RMSTOT  =  SQRT(RMSTOT**2 +  RMSPHS**2)
C      PHTOT  =  PHTOT  +  PMAXAX
       ZLAST  = ZNOW
       IF (INPT .EQ. 0) GO TO 35
       IF(MOD(KM1,INPT).NE.0)GO TO 35
       WRITE (6,33) K,PMAXAX,AXIAL
C      WRITE (6,49) AVELAG,RMSTOT,PHTOT
   33 FORMAT (22H1 FIELD AFTER AXIAL TB,I2,5HPMAX=,G12.5,4HVAX=,G12.5)
C   49 FORMAT(5X,24HTOTAL AVERAGE PHASE LAG ,G12.5,18H TOTAL RMS PHASE= ,
C     XG12.5,26H TOTAL PHASE CHANGE MAX. =,G12.5)
      WRITE (6,44) CAPKAX
   44 FORMAT(10X,10H CAPKAX = ,G12.5)
      N  = 0
      UMAX  = 0.
      CALL OUTPUT(CU,NPY,NPTS,X,N,UMAX,.TRUE.,.FALSE.,.FALSE.)
      GO TO 35
C   *******************   *******    *******  ********************
C   TRANSIENT THERMAL BLOOMING CALCULATIONS ARE DONE IN THIS SECTION.
C   ENERGY EQUATION  IS SOLVED FOR PHASE CHANGE AS A FUNCTION OF
C   BEAM ON TIME.
C   *******************   *******    *******  ********************
  220 ETA =(ALFA * GDC ) / ( T * CP )
      ZNOW = ZLAST + DELZ
      IF (INPT .EQ. 0) GO TO 210
      IF(MOD(KM1,INPT).NE.0)GO TO 210
      WRITE(6,279) DT,ETA,DELZ,ZNOW,XFACT,PRED
  279 FORMAT (/,7H  DT = ,G12.5,7H ETA = ,G12.5,8H DELZ = ,G12.5,/,
     X5H Z = ,G12.5,10H  XFACT = ,G12.5,9H  PRED = ,G12.5)
        WRITE (6,278) K
  278 FORMAT(54H1 FIELD INCIDENT ON TRANSIENT THERMAL BLOOMING ELEMENT
     X,I2)
```

```
      N = 0                                                        TBLOOM
      UMAX = 0.0                                                   TBLOOM
      CALL OUTPUT(CU,NPY,NPTS,X,N,UMAX,.TRUE.,.FALSE.,.FALSE.)     TBLOOM
  210 NWRITE = NPY / 2.                                            TBLOOM
      ZLAST =  ZNOW                                                TBLOOM
      POWER = 0.0                                                  TBLOOM
      FACTOR= ETA * DT * DELZ * 6.2831853 / WL * XFACT             TBLOOM
      DO 300 L = 1, NPY                                            TBLOOM
      J = NPTS*(L-1)                                               TBLOOM
      DO 300 I = 1, NPTS                                           TBLOOM
      IJ = I + J                                                   TBLOOM
C     XIXY = CU(IJ) * CONJG( CU(IJ) )                              TBLOOM
      XIXY = CUR(2*IJ-1)**2 + CUR(2*IJ)**2                         TBLOOM
      DPHI = FACTOR * XIXY                                         TBLOOM
C     DPHI = (ETA * DT * DELZ * 6.2831853 / WL) * XIXY             TBLOOM
C     CU( IJ ) = CU( IJ ) * CEXP( CMPLX (0., DPHI) ) * PRED        TBLOOM
      CU( IJ ) = CU( IJ ) * CMPLX(COS(DPHI),SIN(DPHI)) * PRED      TBLOOM
C 300 POWER  = POWER + CU(IJ)*CONJG(CU(IJ))                        TBLOOM
  300 POWER  = POWER + XIXY                                        TBLOOM
      POWER  = POWER * DXSQ *NPTS/NPY *XFACT                       TBLOOM
      WRITE(6,295) PT,POWER                                        TBLOOM
  295 FORMAT(10X,6H PT = ,G12.5,10H  POWER = ,G12.5)               TBLOOM
      IF (INPT .EQ. 0 ) GO TO 35                                   TBLOOM
      IF(MOD(KM1,INPT).NE.0)GO TO 35                               TBLOOM
      WRITE (6,281) K                                              TBLOOM
  281 FORMAT(49H1 FIELD AFTER TRANSIENT THERMAL BLOOMING SEGMENT ,I2)  TBLOOM
      UMAX = 0.                                                    TBLOOM
      N=0                                                          TBLOOM
      CALL OUTPUT (CU,NPY,NPTS,X,N,UMAX,.TRUE.,.FALSE.,.FALSE.)    TBLOOM
  250 CONTINUE                                                     TBLOOM
C  35 IF (K.LT.NSTEPS) CALL CORE(DELZ,0,M)                         TBLOOM
   35 IF (K.LT.NSTEPS)                                             TBLOOM
     1CALL STEP(DELZ    ,RAD,.1,.1,NST, 0,0,0.,0.,M,1)             TBLOOM
C 100 IF (K.EQ.NSTEPS) CALL CORE(DELZ/2.,IOUT,M)                   TBLOOM
  100 IF (K.EQ.NSTEPS)                                             TBLOOM
     1CALL STEP(DELZ/2.,RAD,.1,.1,NST,IOUT,0,0.,0.,M,1)            TBLOOM
      RETURN                                                       TBLOOM
      END                                                          TBLOOM
```

35. SUBROUTINE THERML

a. Purpose -- Since uncooled mirror glass has such a low coefficient of thermal expansion, the mirror surface heats up as the beam hits it, thus heating up the surrounding boundary layer of air. Subroutine THERML, shown in Figure 70, models the phase change impressed on the beam due to thermal gradients in the boundary layer of air.

b. Relevant formalism -- The theory of this phenomenon was developed by Humphreys and Wick (Ref. 15) of AFWL.

---

15. Humphreys, W. W. and R. V. Wick, "Change in Optical Path Length Near a Hot Mirror Surface," Laser Digest, AFWL-TR-75-140, 1975, p. 9.

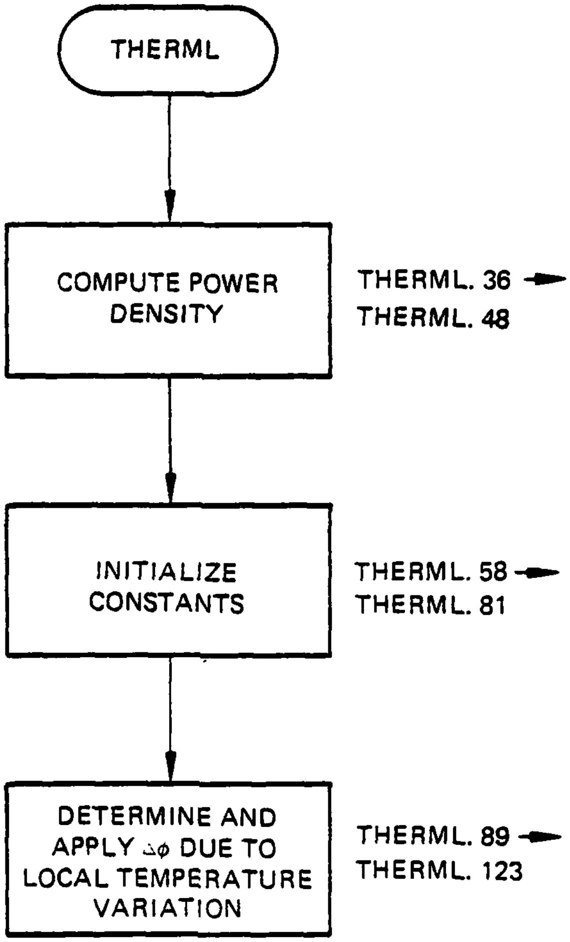


Figure 70. Subroutine THERML organization.

Following Humphreys and Wick, assume that the times of interest are short enough to consider the mirror to be a semi-infinite slab. From the theory of heat conduction the time for heat to traverse a length L is $t = L^2/\alpha$. Thus, for mirrors of thickness L, the time during which the mirror acts like a semi-infinite slab is $\ll L^2/\alpha$. Assume also that for these times one can neglect natural convective cooling. Therefore, the air can also be modeled as a semi-infinite slab. The one-dimensional heat equation is then assumed to apply for both the mirror and the air:

$$\frac{\partial^2 T_m}{\partial x_m} = \frac{1}{\alpha_m}\left(\frac{\partial T_m}{\partial t}\right) \qquad \frac{\partial^2 T_a}{\partial x_a^2} = \frac{1}{\alpha_a}\left(\frac{\partial T_a}{\partial t}\right) \tag{231}$$

Common variable altered:

CU = the field is modified by the boundary layer temperature gradients.

Subroutines called: OUTPUT

where the coordinates are seen in Figure 71.

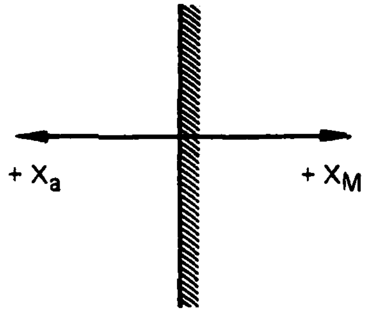


Figure 71. One-dimension heat diagram of mirror and air.

Initially, both the air and the mirror are at the same temperature T_o

$$T_m(x_m,0) = T_o = T_a\ (x_a,0) \tag{232}$$

For the times considered, the heat does not have time to diffuse to the back boundary of either the mirror or the air. This boundary condition can be written

$$T_m(\infty,t) = T_o = T_a\ (\infty,t) \tag{233}$$

The air and the mirror are assumed to maintain the same temperature at their joint boundary so

$$T_m\ (0,t) = T_a\ (0,t) \tag{234}$$

The remaining condition to be applied is that of heat balance at the joint boundary. By Fourier's law

$$-k_m \frac{\partial T_m}{\partial x_m}\bigg|_{x_m = 0} = \alpha I \tag{235}$$

where α is the absorptivity of the mirror. Similarly using Fourier's law at the air boundary

$$-k_a \frac{\partial T_a}{\partial x_a}\bigg|_{x_a = 0} \tag{236}$$

By combining these two equations, the joint heat balance equation at the boundary becomes:

$$-k_m \frac{\partial T_m}{\partial x_m}\bigg|_{x_m = 0} \quad -k_a \frac{\partial T_a}{\partial x_a}\bigg|_{x_a = 0} = \alpha I \tag{237}$$

Since both the media obey the same form of equation, consider the solution of the following equation:

$$\frac{\partial^2 T}{\partial x^2} = \frac{1}{\alpha} \frac{\partial T}{\partial t} \tag{238}$$

Finding the Laplace Transform of the above equation gives

$$\frac{d^2\overline{T}(x,s)}{dx^2} = \frac{1}{\alpha'}\left(-T\ (x,o) + s\overline{T}\ (x,s)\right) \tag{239}$$

where,

$$\overline{T}\ (x,s) = \int_0^{\infty} dt\,\ell^{-st}T(x,t)$$

Noting that $T(x,0) - T_0$ for both the mirror and the boundary layer, one can rewrite this as

$$\frac{d^2}{x^2}\left(\overline{T}(x,s) - \frac{T_o}{s}\right) = \frac{s}{\alpha} * \left(\overline{T}(x,s) - \frac{T_o}{s}\right) \tag{240}$$

which integrates to give

$$\bar{T}(x,s) = \frac{T_o}{s} + A(s)\varepsilon^{\sqrt{\frac{s}{\alpha}}\,x} + B(s)\varepsilon^{\sqrt{\frac{s}{\alpha}}\,x} \tag{241}$$

The boundary condition for $x \to \infty$ implies that $A = 0$ for both media.

Therefore

$$\bar{T}(x,s) - \frac{T_o}{s} = B(s)\varepsilon^{\sqrt{\frac{s}{\alpha}}\,x} \tag{242}$$

To proceed further, it is necessary to determine B(s). This is done using the joint boundary conditions. Recall that

$$-k_m \left.\frac{\partial T_m}{\partial x_m}\right|_{x_m=0} - k_a \left.\frac{\partial T_a}{\partial x_a}\right|_{x_a=0} = \alpha I$$

Assuming (αI) to be constant in time, this transforms to

$$-k_m \left.\frac{\partial \bar{T}_m}{\partial x_m}\right|_{x_m=0} - k_a \left.\frac{\partial \bar{T}_a}{\partial x_a}\right|_{x_a=0} = \frac{\alpha I}{s} \tag{243}$$

but

$$\left.\frac{\partial \bar{T}}{\partial x}(x,s)\right|_{x=0} = -\sqrt{\frac{s}{\alpha}} * \left. B(s)\varepsilon^{\sqrt{\frac{s}{\alpha}}\,x}\right|_{x=0} = -\sqrt{\frac{s}{\alpha}}\;B(s) \tag{244}$$

Therefore

$$-k_m \left(- \sqrt{\frac{s}{\alpha}}\, B_m \right) - k_a \left(- \sqrt{\frac{s}{\alpha}}\, B_a \right) = \frac{\alpha I}{s} \tag{245}$$

Recall that at $x = 0$, $T_m(0,t) = T_a(0,t)$. This implies that $B_m(s) = B_a(s)$.

Therefore

$$B_a = B_m = \frac{\alpha I}{s\sqrt{s}} \; \frac{I}{\dfrac{k_m}{\sqrt{\alpha_m}} + \dfrac{h_a}{\sqrt{\alpha_a}}} \tag{246}$$

The equation for the air to be back-transformed is therefore

$$\overline{T}_a\,(x_a,\, s) - \frac{T_o}{s} = \frac{\alpha I}{\dfrac{km}{\sqrt{\alpha_m}} + \dfrac{k_a}{\sqrt{\alpha_a}}} \; \frac{e}{s\; s}^{-s\,\left(\frac{xa}{\sqrt{\alpha_a}}\right)} \tag{247}$$

Note that $\overline{T}_m\,(x_m,t)$ obeys a similar equation with the a and the m subscripts interchanged. Recall the following Laplace Transform theorems:

$$L(To) = \frac{To}{s}$$

$$\frac{1}{s} L \left(f(t) \right) = L \left(\int_o^t dt\; f(t) \right) \tag{248}$$

and

$$\frac{e^{-a\sqrt{s}}}{\sqrt{s}} = L \left(\frac{e^{-a^2/4t}}{\sqrt{\pi t}} \right) \tag{249}$$

The equation for $T_a(x_a,t)$ is therefore

$$T_a(x_a,t) - T_o = \frac{\alpha I}{\frac{km}{\sqrt{\alpha_m}} + \frac{ka}{\sqrt{\alpha_a}}} \int_o^t \frac{dt \varepsilon^{-\frac{x^2}{4\alpha_a t'}}}{\sqrt{\pi t'}} \tag{250}$$

or

$$\Delta T_a (x_a,t) = T_a (x_a,t) - T_o$$

$$= \frac{\alpha I}{\frac{km}{\sqrt{\alpha_m}} + \frac{ka}{\sqrt{\alpha_a}}} \left\{ 2\sqrt{\frac{t}{\pi}}\, \varepsilon^{-x_a^2/4\alpha_a t} - \frac{x_a}{\sqrt{\alpha_a}} \operatorname{erfc}\left(\frac{x_a}{2\sqrt{\alpha_a t}}\right)\right\} \tag{251}$$

The phase change in the beam induced by this variation in temperature is given by

$$\Delta\emptyset\ (x,y,I) = 2\left(\frac{2\pi}{\lambda}\right) \int_o^{4\sqrt{\alpha_a t}} \left(\frac{dn}{dT_a}\right) \Delta T_a (dx_a) \tag{252}$$

The factor of 2 is due to the fact that the beam passes through the boundary layer twice. The limit on the integral is seen to be the practical point at which the variation in temperature becomes negligible. This limit is important to estimate since the integral is to be done numerically.
As in TBLOOM, dn/dt is found by the Gladstone-Dale law

$$N = 1 + \rho C \tag{253}$$

and the equation of state of a perfect gas

$$\rho = \frac{MP}{RT} \tag{254}$$

at constant pressure

$$\frac{dn}{dt} = \frac{-\rho C}{T} \tag{255}$$

It is assumed that the effect is small enough that the integral may be approximated by a finite number of steps. Four steps are chosen here.

c. Fortran

Argument List

CONMIR = mirror thermal conductivity

CONGAS = boundary layer thermal conductivity

ALPHAM = mirror diffusivity

ALPHAG = boundary layer diffusivity

RHOGAS = boundary layer density

REFMIR = mirror reflectivity

TAU = transient time

TIN = temperature

SUBROUTINE THERML 76/176 OPT=1 FIN 4.6+452 04/27/79 12.23.47

```
      SUBROUTINE THERML(CONMIR,ALPHAM,ALPHAG,RHOGAS,TAU,TIN,REFMIR,
     1CONGAS)
      LEVEL 2, CU,CUR
      COMMON/MELT/CU(16384),CFIL(16512),X(128),WL,NPTS,NPY,DHX,DHY
      COMMON/WAY/WNOW,NREG,NAPTH
      REAL CUR(32768)
      COMPLEX CU,CFIL
      EQUIVALENCE (CUR(1),CU(1))
C     ******************         *********   ********************
C                          E41.E/482A.THERM2
C        THIS ROUTINE CALCULATES THE EFFECT OF A THERMAL BOUNDARY
C         LAYER IN FRONT OF A MIRROR. J. FORGHAM 5 /31 /75
C     ******************         *********   ********************
C
C
C      THIS VERSION CALCULATES PHASE CHANGE BASED ON THE GAS TEMP.
C      RISE IN FRONT OF THE MIRROR ACCORDING TO FORGHAMES SOLUTION
C      AS GENERATED FROM $ HEAT TRANSFER  $ BY HOLMAN.
C
C
C
C                          FORGHAM 5/31/75
C     ******************        ****  *****   ******************
```

```
      WRITE(6,5) ALPHAM,CONMIR,ALPHAG,CONGAS,RHOGAS,TAU,TIN,REFMIR
    5 FORMAT(119H0FIELD HAS ENTERED MIRROR THERMAL BOUNDARY LAYER ROUTIN
     XE.  MEDIUM CONDITIONS                                           /2
     X5X,30HMIRROR DIFFUSIVITY            = ,G12.5,11H CMSQ/SEC     /25X,
     X30HMIRROR THERMAL CONDUCTIVITY = ,G12.5,13H WATT/CM SEC /25X,
     X30HBOUNDARY LAYER DIFFUSIVITY  = ,G12.5,13H CMSQ/SEC    /25X,
     X30HBOUNDARY LAYER THRML CONDUCTIVITY = ,G12.5,13H WATT/CM SEC /25X
     X30HBOUNDARY LAYER DENSITY       = ,G12.5,9H GM/CC   /25X,
     X19HTRANSIENT TIME   = ,G12.5, 7H  SEC  ,/25X,
     X14HTEMPERATURE = ,G12.5,7H DEG. K /25X,
     X14HMIRROR REF. = ,G12.5)
C *** COMPUTE POWER DENSITY
      INPT = 1
      IF (NPTS.GT.32) INPT = 0
      DX = X(2) -X(1)
      DXSQ = DX * DX
      XFACT = 1.
      IF(NNEG .EQ. 1.OR.NNEG .EQ.2) XFACT = 1./NNOW**2
      NOB=NPTS*NPY
      PT = 0.
      DO 10 I=1,NOB
C  10 PT = PT + CU( I )*CONJG(CU( I ))*XFACT
   10 PT = PT + (CUR(2*I-1)**2 + CUR(2*I)**2) * XFACT
      PT = PT*DXSQ*NPTS/NPY
      WRITE(6,14)  PT
   14 FORMAT(46H1 FIELD INCIDENT UPON BOUNDARY LAYER   ELEMENT,  7HPOWER
     1= ,G12.5)
      IF(INPT .EQ. 0) GO TO 15
      N  = 0
      UMAX  = 0.
      CALL OUTPUT(CU,NPY,NPTS,X,N,UMAX)
C     ******************     **********          **************
   15 CONTINUE
C *** INITIALIZE CONSTANTS  ***
C *** ALPHAG THERMAL DIFFUSIVITY OF GAS IN BDY LAYER
C *** ALPHAM THERMAL DIFFUSIVITY OF MIRROR MATERIAL
      PI  = 3.14159
      GOC = .223
      EABS = (1. - REFMIR)
      WN  = (2.* PI)/WL
      NZ = 4
      NZ1 = NZ + 1
      DZ = ( 4.* SQRT(ALPHAG * TAU) )/NZ
      SALFA = SQRT(ALPHAG)
      SALFM = SQRT(ALPHAM)
      C1 = 1./(4.*ALPHAG*TAU)
      C2 = SQRT(C1)
      C3 = 2.*SQRT( TAU/PI )
      C4 =(EABS/CONMIR)*SQRT(PI*ALPHAM*TAU)
      BIGPHI = -100000.
      WRITE(6,2192)SALFM,SALFA,CONGAS,DZ,C1,C2,C3,C4
 2192 FORMAT(10X,23H SALFM SALFA CONGAS DZ ,4G12.5,//,10X,12HC1 C2 C3 C4
     1 4G12.5)
      WRITE(6,1002) EABS,WN
 1002 FORMAT(10X,14HMIRROR ABS =  ,G12.5,11H WAVE NO = ,G12.5)
C  *** FIND DN / DTEMP ***
      DNDT = (-RHOGAS / TIN )* GOC
      WRITE(6,1004) DNDT
 1004 FORMAT(10X,9H DNDT  = ,G12.5)
      IF (INPT.EQ.0)GO TO 1014
      WRITE(6,1005)
 1014 CONTINUE
 1005 FORMAT(10X,52H       X                           Y              DPHIXY
     X)
C  *** FIND LOCAL TEMPERATURE  AND MODIFY FIELD BY THERMAL LENS ***
      IJ = 0
      DO 400 K = 1,NPY
      J = (K - 1) * NPTS
      YY=(K-1) * DX + DX/2.
      DO 400 I = 1,NPTS
      TOTN = 0.0
      IJ = I + J
      XX=(I-1) * DX + DX/2.
```

```
C       XIXY = CU(IJ) * CONJG(CU(IJ))
        XIXY = CUR(2*IJ-1)**2 + CUR(2*IJ)**2
        DO 325 MM = 1,N21
        ZBL=(MM - 1)*DZ
        ARG1 = -C1 * ZBL * ZBL
        ARG2 = C2 * ZBL
        F2 =  ERFC( ARG2 )
        DELT = XIXY    * C4 * F2
        TOTN = TOTN + DELT*UNDT*DZ
  325 DPHIXY =-TOTN * WN *2.
C       IF(INPT .EQ. 0) GO TO 330
        IF(DPHIXY.LT.BIGPHI) GO TO 330
        BIGPHI = DPHIXY
        XIMAX  = XIXY
        XMAX   = XX
        YMAX   = YY
        DELTMX =DELT
C       F1MX   =F1
        F2MX   =F2
        TOTNMX = TOTN
  330 CONTINUE
        IF(INPT .EQ. 0.OR.NPTS .GT.32) GO TO 395
        WRITE(6,1006)XX,YY,DPHIXY
 1006 FORMAT(10X,3(10X,G12.5))
C 395 CU(IJ) = CU(IJ) * CEXP(CMPLX(0.,DPHIXY))
  395 CU(IJ) = CU(IJ) * CMPLX( COS(DPHIXY),SIN(DPHIXY) )
  400 CONTINUE
C       IF(INPT.EQ.0) GO TO 35
        WRITE(6,2913) BIGPHI,XIMAX,XMAX,YMAX,DELTMX,F2MX,TOTNMX
 2913 FORMAT(10X,14H DPHI,IMAX,X,Y,4G12.5,//,18H DTEMP,F1,F2,DELN ,3G12.
     X5)
        IF(INPT.EQ.0) GO TO 35
   34 FORMAT(61H1 FIELD AFTER MODIFICATION BY THERMAL BOUNDARY LAYER ELE
     1MENT )
       N  = 0
       UMAX  = 0.
       CALL OUTPUT(CU,NPY,NPTS,X,N,UMAX)
   35 RETURN
        END
```

36. SUBROUTINE TILT

a. Purpose -- Subroutine TILT, shown in Figure 72, can be used to remove beam tilt and will calculate the radius of curvature of a beam.

b. Relevant formalism -- To remove small amounts of beam tilt, the following formalism is used. Large fixed tilts, such as result from mirrors set at an angle to the beam axis, are removed by the system analyst in defining the equivalent collimated system.

Consider an input field $U(x,y)$ incident on an optical element with transmission function $t(x,y)$ yielding an output $U'(x,y)$.

$$\begin{aligned} U'(x,y) &= t(x,y)\, U(x,y) \\ &= A \exp(i\phi) \end{aligned} \tag{256}$$

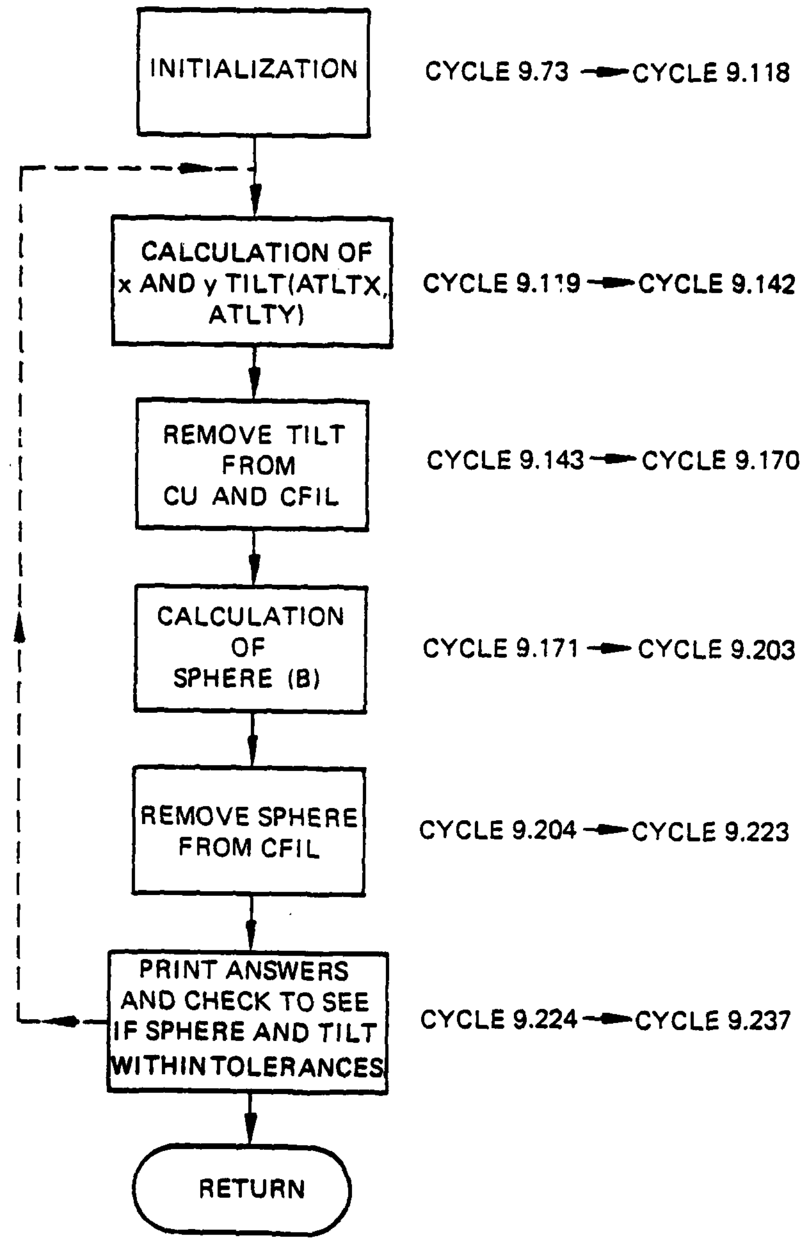


Figure 72. Subroutine TILT organization.

For removal of beam tilt from a field U(x,y) the transmission function must be of the form

$$t_{TILT}(x,y) = e^{-i(a_x x + a_y y)} = e^{-i\vec{a}\cdot\vec{X}} \tag{257}$$

where $a_x \simeq h\Theta_x$ and $A_y \simeq h\Theta_y$ define the tilt angles to be removed.

Similarly, the phase curvature is removed by the following transmission function

$$t_{SPHERE}(x,y) = \varepsilon^{-i\frac{k}{2R}(x^2+y^2)} \tag{258}$$

To calculate the constants a_x and a_y for an arbitrary field distribution, U(x,y), define the following functional to be minimized:

$$F_{TILT} = \iint dxdy \left|U(x,y)\right|^2 \left[\nabla(\phi - a_x\ x - a_y y\right]^2 \tag{259}$$

or

$$F_{TILT} = \iint dxdy \left|U(x,y)\right|^2 \left[\left(\frac{\partial\phi}{\partial x} - a_x\right)^2 + \left(\frac{\partial\phi}{\partial y} - a_y\right)^2\right]$$

the resulting expression for $\vec{a}$ is

$$\vec{a} = \langle\vec{\nabla}\phi\rangle \tag{260}$$

where,

$$\langle\vec{\nabla}\phi\rangle = \frac{\iint d\vec{x}\ \left|U\ (\vec{x})\right|^2 \vec{\nabla}\phi}{\iint d\vec{x} \left|U\ (\vec{x})\right|^2} \tag{261}$$

7ϕ is easily found from the field data by noting that

$$\mathrm{Im}\ (U^*\vec{\nabla}U) = \left|U\right|^2\vec{\nabla}\phi \tag{262}$$

Once the tilt is removed, a similar procedure to remove phase curvature is used. Recall that the transmission function t_{SPHERE} (x,y) needed is of the form

$$t_{SPHERE}(x,y) = \varepsilon^{-ik\left(\frac{x^2+y^2}{2R}\right)} \tag{263}$$

The new functional to be minimized is

$$F_{SPHERE} = \iint dxdy \left| U(x,y) \right|^2 \left[\nabla \left(\phi - b\left(\frac{x^2 + y^2}{2}\right)\right)\right]^2 \tag{264}$$

which results in

$$b = \frac{\langle \vec{x} \cdot \vec{7}\mathfrak{z} \rangle}{\langle \vec{x} \cdot \vec{x} \rangle} \tag{265}$$

Valves of tilt a and sphere b are found by an iterative procedure until the values established for these parameters do not change appreciably.

c. Fortran

Argument List

AX, AY = Total x and y tilt in the beam. The amount of tilt removed from the beam by this routine is added to these parameters so that no tilt information is lost.

RADCUR = the negative of the radius of curvature of the beam found by this routine. To produce a "flat" beam the following calculation would be performed.

$$CU'(I,J) = CU(I,J) * \exp\ i(\pi/\lambda R)\ (x^2 + y^2) \tag{266}$$

with R representing RADCUR

X = X(I) and Y = X(J)

IPS = the parameter that indicates which options in this routine are to be used. IPS is the same parameter as IIPS in name list PROPGT in subroutine GDL. The options are:

IPS = 0 Tilt is not called for

= 1 Tilt only is removed

= 2 Sphere only found

= 3 Both tilt and sphere found, tilt being removed.

Common Variables Altered

CU - has tilt removed

CFIL - starts off set to CU, then has both tilt and sphere removed.

Subroutine TILT computer printouts follow.

SUBROUTINE TILT 76/176 OPT=1 FIN 4.6+452 04/27/79 12.23.47

```
      SUBROUTINE TILT(AX,AY,RADCUR,IPS)                                 CYCLE9
C     PHASE CORRECTION ROUTINE                                          CYCLE9
C     THIS ROUTINE DETERMINES THE LINEAR AND QUADRATIC COMPONENTS OF    CYCLE9
C     PHASE.  IT ALSO REMOVES THE LINEAR COMPONENT BEFORE RETURNING     CYCLE9
C     TO THE CALLING ROUTINE.                                           CYCLE9
      LEVEL 2, CU,CUR,CFILR                                             CYCLE9
      COMMON /MELT/ CU(16384),CFIL(129,128),X(128),WL,NPTS,NPY,ORX,ORY  CYCLE9
      COMPLEX CU,CFIL,CSUMX,CSUMY,CB,CA,CAA,CCC,CC,CCNJ,CAX,CAY,CFACT,  CYCLE9
     A CHX,CHY,CCNJ,CHINT                                               CYCLE9
      DIMENSION CUR(1),CFILR(258,1)                                     CYCLE9
      EQUIVALENCE (CU(1),CUR(1)) , (CFIL(1,1),CFILR(1,1))               CYCLE9
      WRITE(6,301)                                                      CYCLE9
  301 FORMAT(70H0*** LINEAR AND/OR SPHERICAL COMPONENTS OF PHASE ARE BEI CYCLE9
     XNG REMOVED ***/)                                                  CYCLE9
      ITMAX=30                                                          CYCLE9
      SPHTOL=.001                                                       CYCLE9
      ICKA=0                                                            CYCLE9
      ICKR=0                                                            CYCLE9
      PI=3.141592                                                       CYCLE9
      DELX = X(2)-X(1)                                                  CYCLE9
      AXTOT=0.0                                                         CYCLE9
      AYTOT=0.0                                                         CYCLE9
      KOUNT = 0                                                         CYCLE9
      EAX = 0.0                                                         CYCLE9
      EAY = 0.0                                                         CYCLE9
      ERP = 0.0                                                         CYCLE9
      RADCUR = 1.E50                                                    CYCLE9
      ROOLD = 1.E50                                                     CYCLE9
      AXOLD = AX                                                        CYCLE9
      AYOLD = AY                                                        CYCLE9
      POW = 0.0                                                         CYCLE9
      DO 20 J=1,NPY                                                     CYCLE9
      DO 20 I=1,NPTS                                                    CYCLE9
      IJ = I + (J-1)*NPTS                                               CYCLE9
      CFIL(I,J) = CU(IJ)                                                CYCLE9
      POW = CUR(IJ*2-1)**2 + CUR(IJ*2)**2  + POW                        CYCLE9
```

```
C     POW = CFIL(I,J)*CONJG(CFIL(I,J))+POW
   20 CONTINUE
      POW = POW*DELX**2
      NLIMX = NPTS-1
      NLIMY = NPY-1
      IF (NPTS.NE.NPY)NLIMY=NPY
      WRITE(6,180)
  180 FORMAT (27X,33HINTERMEDIATE OPTIMIZATION RESULTS //
     A 10H ITERATION,5X,5HFOCAL,7X,6HRADCUR,9X,3HATX,10X,3HATY,
     B 8X,5HAXTOT,8X,5HAYTOT    )
   25 IF (IPS .EQ. 2 ) GO TO 54
      KOUNT = KOUNT+1
      CSUMX = (0.0,0.0)
      CSUMY = (0.0,0.0)
      DO 30 J=2,NLIMY
      J1=J+1
      JM=J-1
      IF(J.EQ.NPY)J1=J
      CB=CFIL(1,J)
      CA=CFIL(2,J)
      DO 30 I=2,NLIMX
      CAA=CFIL(I,J1)
      CCC=CFIL(I,JM)
      CC=CB
      CB=CA
      CA=CFIL(I+1,J)
      CCNJ = CONJG(CB)
      CSUMX = CCNJ*(CA-CC)/2.0+CSUMX
      CSUMY = CCNJ*(CAA-CCC)/2.0+CSUMY
   30 CONTINUE
      CAX = CSUMX*DELX
      CAY = CSUMY*DELX
      ATLTX =AIMAG(CAX)/POW
      ATLTY =AIMAG(CAY)/POW
      IF(NPTS.EQ.NPY) GO TO 52
      ATLTY=0.0
   52 ATX=-ATLTX*WL/(2.*PI)
      ATY=-ATLTY*WL/(2.*PI)
      AXTOT=AXTOT+ATX
      AYTOT=AYTOT+ATY
      AX=AX+ATX
      AY=AY+ATY
      DO 40 J=1,NPY
      J1=(J-1)*NPTS
      ATLTYY = ATLTY * X(J)
      DO 40 I=1,NPTS
      INDX=J1+I
      PHI = ATLTX*X(I) + ATLTYY
      CFACT = CMPLX (COS(PHI),SIN(PHI))
C     CFACT = CEXP(CMPLX(0.,ATLTX*X(I)+ATLTY*X(J)))
      CU(INDX)=CU(INDX)/CFACT
      CFIL(I,J)=CFIL(I,J)/CFACT
   40 CONTINUE
      EAX = 0.0
      EAY = 0.0
      IF(ABS(AX) .GT. 0.0)EAX=ABS(1.0-AXOLD/AX)
      IF(ABS(AY) .GT. 0.0)EAY=ABS(1.0-AYOLD/AY)
      ICKA = 1
      IF(EAX.LT.0.05.AND.EAY.LT.0.05)ICKA=0
      AXOLD = AX
      AYOLD = AY
      IF (IPS .EQ. 1 ) GO TO 70
C   ******************************************************************
C   * THE FOLLOWING CALCULATIONS DETERMINE THE LEAST SQUARES SPHERICAL*
C   * FIT TO THE PHASE GRADIENT.----THE RESULT ,B, IS 2*PI /(WL*R).  *
C   * R = THE RADIUS OF CURVATURE OF THE PHASE FRONT.                *
C   ********************** FORGHAM 11/25/74 **************************
   54 T= 0.0
      DO 55 J = 1, NPY
      DO 55 I = 1, NPTS
      FMAG = CFILR(2*I-1,J)**2 + CFILR(2*I,J)**2
```

```
C      FMAG = CFIL(I,J) * CONJG( CFIL(I,J) )
       T  =  FMAG * ( X(I)**2 + X(J)**2 ) + T
    55 CONTINUE
       TINT = T * DELX * DELX
       CHX = (0.,0.)
       CHY = (0.,0.)
       DO 60 J = 2 ,NLIMY
       J1=J+1
       JM=J-1
       IF(J.EQ.NPY)J1=J
       CB=CFIL(1,J)
       CA=CFIL(2,J)
       DO 60 I=2,NLIMX
       CAA=CFIL(I,J1)
       CCC=CFIL(I,JM)
       CC=CB
       CB=CA
       CA=CFIL(I+1,J)
       CCNJ = CONJG(CB)
       CHX = CCNJ*(CA-CC)*X(I)/2.0+CHX
       CHY = CCNJ*(CAA-CCC)*X(J)/2.0+CHY
    60 CONTINUE
       CHINT = DELX * ( CHY + CHX )
       B  =  -AIMAG( CHINT ) / TINT
       IF(ABS(B).GT.(2.*PI/WL/1.E50)) FOCAL = 2*PI/(WL*B)
       RADCUR = (FOCAL*RADCUR)/(FOCAL+RADCUR)
       IF(ABS(RADCUR).GT.0.0)ERD=ABS(1.0-RDOLD/RADCUR)
       ICKR=1
       RDOLD = RADCUR
       IF(ERD.LE.SPHTOL)ICKR=0
C      CRAD = CMPLX(0.0,PI/(WL*FOCAL))
       PIOWLF = PI/(WL*FOCAL)
       DO 80 J=1,NPY
        YSQ  = X(J)**2
       DO 80 I=1,NPTS
        I2   = 2*I
        I2M1 = I2 - 1
        PHI  = (X(I)**2 + YSQ) * PIOWLF
        SINP = SIN(PHI)
        COSP = COS(PHI)
        CURS = CFILR(I2M1,J)
        CFILR(I2M1,J) = CURS*COSP - CFILR(I2,J)*SINP
    80  CFILR(I2,J)    = CURS*SINP + CFILR(I2,J)*COSP
C   80 CFIL(I,J)=CFIL(I,J)*CEXP((X(I)**2+X(J)**2)*CRAD)
    70 UMAX = 0.0
       WRITE(6,190) KOUNT,FOCAL,RADCUR,ATX,ATY,AXTOT,AYTOT
   190 FORMAT(1X,I5,4X,6G13.4)
       IF(FOCAL.GT.-4.E05.AND.FOCAL.LT.6.E05.AND.KOUNT.LT.ITMAX)GO TO 25
       IF((ICKA.GT.0.OR.ICKR.GT.0).AND.KOUNT.LT.ITMAX) GO TO 25
       IF(IPS.EQ.1.OR.IPS.EQ.3)WRITE(6,201)AXTOT,AYTOT
   201 FORMAT(/20X,16HLINEAR COMPONENT//
      X     10X,8HTILT IN 1HX9H = A(X) =,G12.4,8H RADIANS/
      X     10X,8HTILT IN 1HY9H = A(Y) =,G12.4,8H RADIANS)
       IF(IPS.GE.2)WRITE(6,67)RADCUR
    67 FORMAT(/20X,19HSPHERICAL COMPONENT//
      X     10X,32HPHASE FRONT CURVATURE = RADCUR =,G12.4,3H CM//)
       RETURN
       END
```

37. SUBROUTINE ERF

a. Purpose -- The function ERF generates the error function

$$\mathrm{erf}(x) = \frac{2}{\sqrt{\pi}} \int_0^x \epsilon^{-t^2} dt \qquad (267)$$

or its complement, 1-erf(x), for any input value of x. This subroutine is a copy of the ERF function available from the AFWL scientific program library. Figure 73 shows the Subroutine ERF flow chart.

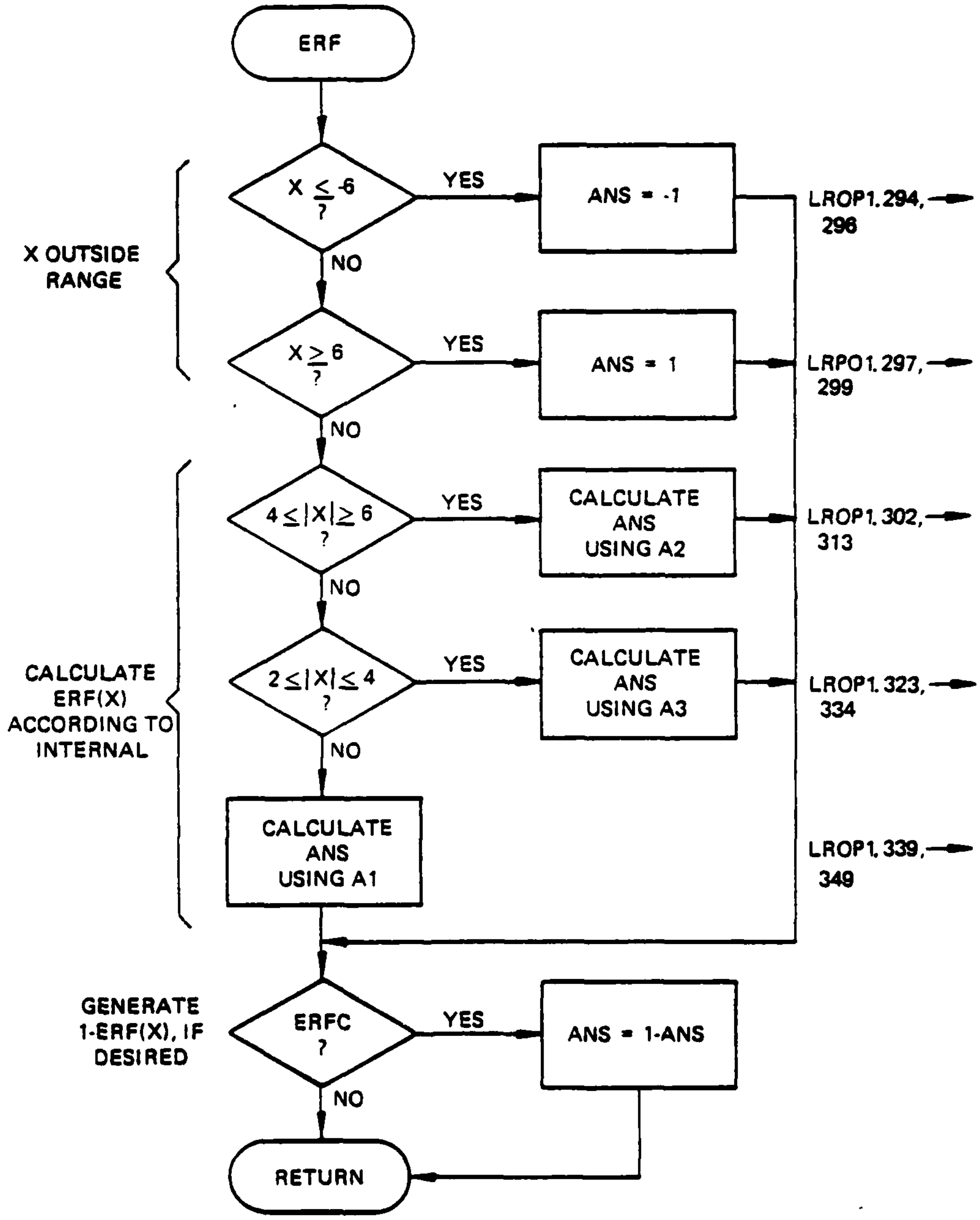


Figure 73. Subroutine ERF flow chart.

b. Relevant formalism -- The error function integral is approximated over discrete intervals of the argument, x, by Tchebickef (Chebychev) polynomials. These polynomials are evaluated in a loop which combines the recurrence relations for generating the polynomials and a running summation of the terms as they are generated. Coefficients for the polynomials are provided in a data statement for three discrete ranges of the argument. Argument values outside this range will return a zero (0).

Argument List

ANS	error function value returned to calling program
KODE	flag to indicate computation of erf(x) or 1-erf(x)
XX	error function argument

Relevant Variables

A1	array of coefficients used in the polynomial expansion over the range $\lvert xx \rvert \leq 2.$
A2	coefficient array for the range $4. \leq \lvert xx \rvert \leq 6.$
A3	coefficient array for the range $2. < \lvert xx \rvert < 4.$

SUBROUTINE ERF 76/176 OPT=1 FTN 4.6+452 04/27/79 12.23.47

```
      SUBROUTINE ERF(KODE,XX,ANS)
C     COMPUTES BY CHEBYSHEV EXPANSIONS ON INTERVALS.
C     KODE=1  COMPUTES ERF(X)
C     KODE=2  COMPUTES ERFC(X)=1-ERF(X)
C
C     CDC 6600 ROUTINE
C     1-4-72
C
      DIMENSION A1(31),A2(27),A3(16)
      DATA(A1(I),I=1,31)/2.96622112816961E+0,0.,-6.02142146773189E-1,0.,
     11.37989661379662E-1,0.,-2.78325425294437E-2,0.,4.86159904486783E-3
     2,0.,-7.31727931694 53E-4,0.,9.72419688637174E-5,0.,-1.14985131161 8
     304E-5,0.,1.22264871646933E-6,0.,-1.17982030973170E-7,0.,1.04140177
     46912 78E-8,0.,-8.46595329454225E-10,0.,6.37620443498960E-11,0.,-4.4
     5717728196221 5E-12,0.,2.93540222982101E-13,0.,-1.83283038964141E-14
     6/
C
      DATA(A2(I),I=1,27)/1.97070527225754,0.,-1.43397402717750E-2,0.,
     12.97361692202619E-4,0.,-9.80351604336237E-6,0.,4.33133420347280E-7,
     20.,-2.362150026241E-8,0.,1.51549676581E-9,0.,-1.10849398506E-10,
     30.,9.04259014E-12,0.,-8.0947054E-13,0.,7.853856E-14,0.,
     4-8.17918E-15,0.,9.0715E-16,0.,-1.0646E-16/
C
      DATA(A3(I),I=1,16)/1.06630885319 93E+0,1.78876062094436E-2,-3.8017
     15293804401E-3,6.97111435023601E-4,-1.16368846063892E-4,1.81367675 9
     232619E-5,-2.67719439785138E-6,3.77701329909996E-7,-5.12491142501 40
     32E-8,6.71870395763107E-9,-8.54019646112644E-10,1.05544302186899E-1
     40,-1.27108499000124E-11,1.49441348185064E-12,-1.71382907805335E-13
     5,2.08849564313469E-14/
```

```
C
      DATA HTPI,XLIM/1.77245385090552, 2.58408528684382E+1/
      DATA N1,N1M1,N2,N2M1,N3,N3M1/31,30,27,26,16,15/
C
      X=XX
      GO TO (100,200),KODE
  100 CONTINUE
      IF(X.LE.-6.)  10,20
   10 ANS=-1.
      RETURN
   20 IF(X.LT.6.) GO TO 12
      ANS=1.
      RETURN
   12 IF(X.LT.4.) GO TO 30
      ASSIGN 26 TO ISET
   61 CONTINUE
      Z=4./X  $  TZ=Z+Z
      B2=0.
      B1=0.
      DO 25 I=1,N2M1
      J=N2-I+1
      TEMP=B1
      B1=TZ*B1-B2+A2(J)
      B2=TEMP
   25 CONTINUE
      ANS=Z*B1-B2+A2(1)/2.
      ANS=(EXP(-X*X)/(X*HTPI))*ANS
      GO TO ISET,(26,27,28,29)
   26 ANS=1.-ANS
   27 RETURN
 29   ANS = -1.+ANS
      RETURN
   30 CONTINUE
      IF(X.GT.2.) 31,40
   31 CONTINUE
      ASSIGN 26 TO ISET
   35 CONTINUE
      Z=X-3.  $ TZ = Z+Z
      B2=0.
      B1=0.
      DO 36 J=1,N3M1
      K=N3-J+1
      TEMP=B1
      B1=TZ*B1-B2+A3(K)
      B2=TEMP
   36 CONTINUE
      ANS=Z*B1-B2+A3(1)/2.
      ANS=EXP(-X*X)*ANS/X
      GO TO ISET,(26,27,28,29)
   40 CONTINUE
      IF(X.LT.-2.) GO TO 50
      ASSIGN 27 TO ISET
   42 CONTINUE
      Z=X/2.  $  TZ=Z+Z
      B2=0.
      B1=0.
      DO 45 I=1,N1M1
      J=N1-I+1
      TEMP=B1
      B1=TZ*B1-B2+A1(J)
      B2=TEMP
   45 CONTINUE
      ANS=(X/2.)*(Z*B1-B2+A1(1)/2.)
      GO TO ISET,(26,27,28,29)
   50 CONTINUE
      IF(X.GT.-4.) 51,60
   51 CONTINUE
      ASSIGN 29 TO ISET  $  X=-X  $  GO TO 35
   60 CONTINUE
      X=-XX
      ASSIGN 29 TO ISET
      GO TO 61
```

```
  200 CONTINUE
      IF(X.GT.-6.) GO TO 205
      ANS=2.  $  RETURN
C
  205 IF(X.LT.XLIM) GO TO 210
      ANS=0.  $  RETURN
  210 CONTINUE
      IF(X.LT.4) GO TO 215
      ASSIGN 27 TO ISET
      GO TO 61
C
  215 IF(X.GT.2.) GO TO 220
      IF(X.LT.-2.) GO TO 225
      ASSIGN 26 TO ISET
      GO TO 42
C
  220 ASSIGN 27 TO ISET
      GO TO 35
C
  225 IF(X.GT.-4.) GO TO 230
      ASSIGN 28 TO ISET
      X=-XX  $  GO TO 61
   28 ANS=2.-ANS  $  RETURN
C
  230 ASSIGN 28 TO ISET  $  X=-X
      GO TO 35
      END
```

SECTION IV

USER FAMILIARIZATION PACKAGE

The following section contains sample input to run the SOQ code and to logically define the sequence of input to model a sample resonator or optical train the following examples are included:

1. Propagate for Users Guide - Camp
2. Propagate for Users Guide - Vamp
3. Quality for Users Guide
4. Design of a Bare Confocal Resonator
5. Resonator for Users Guide - Bare
6. Resonator for Users Guide - Loaded
7. Sample Code Update

1. PROPAGATE FOR USERS GUIDE - CAMP

```
JRAPC,SIMFX,P4000,T177,EC1.  PROPAGATE FOR USERSGUIDE - CAMP
ACCOUNT(JRALT,********-***,LRO,1731)
GETPF(OLDPL,SOQ77128,ID=*******)
UPDATE(F,W)
FTN(I,LCM=I,PL=20000,L=0,A)
RETURN(OLDPL)
COPYCR,INPUT,TAPE5.
REWIND,TAPE5.
RFLFC(430)
LGO(PL=60000)
RFLFC(1)
*EOR
       PROPAGATE - CAMP
  $START  WWL=0.00106, NCALL=2, DCAL=15., NNPTS=128,
    IH=8, DDRX=0.0, DDRY=0.0, AMPGES=20.0, DGAUSS=0.0,
    RESTRT=.FALSE., PLOTS=1.0, IN=5,
    SYMTRC=.FALSE., PHIRAD=0.0, $END
       PROPAGATE - CAMP
  $CONTRL IFLOW=4.  $END
      APERTURE THE PLANE WAVE TO 10. CM.
  $APTUR  DOUT=10.,  DIN=0.,  $END
  $CONTRL IFLOW=8.  $END
      PLOT THE INITIAL PLANE WAVE
  $PLOT  $END
       INITIAL PLANE WAVE
  $CONTRL  IFLOW=3.  $END
    PROPAGATE THE FIELD 4000 CM.  USING CONSTANT AREA MESH
  $PROPGT  DFLZ=4000., ROCURV=0.,  WINDOX=0.1, WINDOK=0.1,
```

```
            IIFG=1,     IITR=0,      IIPS=0,      $END
  $CONTRL  IFLOW=8,  $END
      PLOT PROPAGATED FIELD
  $PLOT  $END
     PROPAGATED FIELD
  $CONTRL  IFLOW=9,  $END
     RETURN TO MAIN
  $START  WWL=-1.,   $END
*EOR
*EOF
```

2. PROPAGATE FOR USERS GUIDE - VAMP

```
100=JRAUG,STMFX,P60,T77,EC1. PROPAGATEFORUSERSGUIDEVAMP, ID=LREPPEF
110=ACCOUNT(JRALT,00011498-1EL,LRO,1487)
120=ATTACH(OLDPL,SOQ77128, ID=LROPJRA, ST=ANY)
130=UPDATE(F)
140=FTN(I,LCM=I,PL=20000,L=0)
150=RETURN(OLDPL)
160=COPY, INPUT, TAPE5.
170=REWIND, TAPE5.
180=RFLEC(430)
190=LGO(PL=60000)
200=RFLEC(1)
210=*EOR
220=*EOR
230= PROPAGATE A MIRRORED PLANE WAVE A DISTANCE DELZ - VAMP
240=   $START WWL=0.00106, NCALL=2, DCAL=5.6, NNPTS=128,
250=   IB=8, DDRX=0.0, DDRY=0.0, AMPGES=20.0, DGAUSS=0.0,
260=   RESTRT=.FALSE., PLOTS=1.0, IN=5,
270=   SYMTRC=.FALSE., PHIRAD=0.0,  $SEND
280=   PROPAGATE A MIRRORED PLANE WAVE A DISTANCE DELZ - VAMP
290= $CONTRL IFLOW=2,  $SEND
300= APPLY A MIRROR TO THE PLANE WAVE
310= $MIROR DIAOUT=4.0, DIAIN=0.0, XMPOS=0.0, YMPOS= 0.0,
320= RADC=-400., RMIR=1., =$SEND
330= $CONTRL IFLOW=8,  $SEND
340=   PLOT THE MIRRORED PLANE WAVE FIELD
350= $PLOT  $SEND
360= INITIAL MIRRORED PLANE WAVE FIELD
370= $CONTRL IFLOW=3,  $SEND
380= PROPAGATE THE FIELD 200. CM. USING VARIABLE AREA MESH
390= $PROPGT DELZ=200., RDCURV=0., WINDOX=0.1, WINDOK=0.1,
400= IIFG=2, IITR=1, IIPS=0,  $SEND
410= $CONTRL IFLOW=8,  $SEND
420= PLOT PROPGATED FIELD
430= $PLOT  $SEND
440= PROPAGATED FIELD
450= $CONTRL IFLOW=9,  $SEND
460= RETURN TO MAIN PROGRAM
470= $START WWL=-1.,  $SEND
480=*EOR
490=*EOF
```

3. QUALITY FOR USERS GUIDE

```
JRADU,SIMPX,P4000,1177,ECL.    QUALITY FOR USERS GUIDE
ACCOUNT(JRALT,********-***,LRO,1731)
GETPF(OLDPL,S0077128,ID=*******)
GETPF(TAPE8,USERSGUIDEBARECU,ID=*******)
UPDATE(F,W)
FTN(I,LCM=I,PL=20000,L=0,A)
RETURN(OLDPL)
COPYCR,INPUT,TAPE5.
REWIND,TAPE5.
RFLEC(430)
LGO(PL=60000)
RFLEC(1)
*EOR
  FIND THE QUALITY OF THE  FIELD
  $START  WWL=0.00106, NCALL=2, JCAL=13.78, NNPTS=128,
    IB=8, DORX=0.0, DORY=0.0, AMPGFS=1.0, DGAUSS=0.0,
    RESTRT= .TRUE., PLOTS=1.0, IN=5,
    SYMTRC=.FALSE., PHIRAD=0.0, $END
  FIND THE QUALITY OF THE  FIELD
  $CONTRL IFLOW=8,   $END
        PLOT THE  FIELD
  $PLOT    $END
             FIELD AT INPUT
  $CONTRL IFLOW=9,   $END
       RETURN TO MAIN PROGRAM FOR QUALITY CALCULATION
  $START NCALL=3,          $END
             DB = 10.64
  $QLOT DB=10.64,   ISAV=0,IQLT=0, IPHASE=3,    $END
             DB = 10.64
  $START WWL=-1.,   $END
*EOR
*EOF
```

4. DESIGN A BARE CONFOCAL RESONATOR

Assume that one wishes to design a positive branch, unstable bare resonator with a collimated output beam for a given geometric coupling C_g, length L, and concave mirror size (a_1). To solve this problem design a confocal resonator in the following fashion:

Geometric Resonator Design (Fig. 74).

Define the following parameters

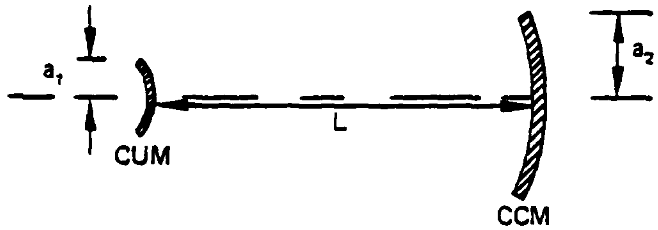


Figure 74. Geometric resonator design.

Recall the definition of geometric coupling.

$$C_g = \frac{A_{OUT}}{A_{TOTAL}} = \frac{\pi a_2^2 - \pi a_1^2}{\pi a_2^2} = 1 - \frac{1}{\left(\frac{a_2}{a_1}\right)^2} \tag{268}$$

But $M = a_2/a_1$ is the magnification of the resonator, thus

$$Cg = Cg = 1 - \frac{1}{M^2} \tag{269}$$

Or inverting this expression, one finds

$$M = \frac{1}{\sqrt{1-C_g}} \tag{270}$$

Given the magnification and length of the resonator, one can find the required mirror radii of curvature, since for an aligned confocal resonator both the convex and concave mirror foci are coincident. Figure 75 describes this coincident feature.

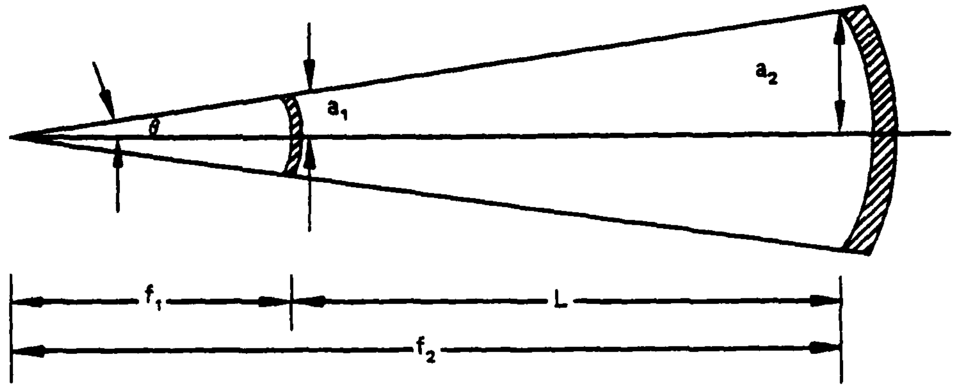


Figure 75. Required mirror radii of curvature.

a. The focal lengths can be related to the magnification by noting that

$$\tan\theta = \frac{a_1}{f_1} = \frac{a_2}{f_2} = \frac{a_3}{f_3} \tag{271}$$

therefore

$$M = \frac{a_2}{a_1} = \frac{f_2}{f_1} = \frac{f_1+L}{f_1} \tag{272}$$

The focal lengths are then found to be

$$f_1 = \frac{L}{M-1} \quad \text{and} \quad f_2 = Mf_1 = \frac{ML}{M-1} \tag{273}$$

Since the radius of curvature of a mirror is twice its focal length, the two radii of curvature are

$$R_1 = \frac{-2L}{M-1} \qquad R_2 = -MR_1 = \frac{2ML}{M-1} \tag{274}$$

where the negative sign indicates a convex mirror and the positive, a concave. For example, if L = 200 cm and C_g = 0.75, the magnification and radii are found to be

$$M = \frac{1}{\sqrt{0.25}} = 2 \tag{275}$$

$$R_1 = \frac{-(2)(200)}{(1)} = -400 \text{ cm} \quad \text{and } R_2 = -(2)(-400) = 800 \text{ cm} \tag{276}$$

b. Tube Fresnel number -- The tube Fresnel number for this resonator can be found by the fact that the expanding pass propagation distance L has an equivalent collimated propagation length of ML so the round trip collimated propagation distance is (M + 1)L. The tube Fresnel number is then (assuming the CVM is 2.0 cm in radius and the beam has a wave length of 10.6 μm).

$$N_T = \frac{a_1}{(M+1)L\lambda} = \frac{(2)^2}{(3)(200)(10.6\times10^{-4})} = 6.29 \tag{277}$$

c. Computer requirements

(1) Overlap -- Since the beam diffracts during propagation, it is necessary to have a large enough calculation region to always contain the beam. The required overlap can be calculated according to Sziklas and Siegman (Ref. 2) as

$$G \geq 1 + \frac{1}{2\pi^2 N_T \varepsilon} \tag{278}$$

where ε is the tolerance on fractional energy loss during propagation. Taking this to be 0.02, one find the guardband to be

$$G \geq 1 + \frac{1}{2\pi^2(6.29)(0.02)} = 1.4 \tag{279}$$

Thus the initial calculation region must be at least G times the beam size:

$$\text{DCALC} = 1.4 \times 2\text{x}2 = 5.6 \text{ cm} \tag{280}$$

(2) Number of points required -- Sziklas and Siegman also show that in order to adequately sample the beam, the number of points in each dimension must obey the following inequality:

$$N_p \geq 4G(G + 1)\ N_T \tag{281}$$

This becomes

$$N_p \geq 4(1.4)(2.4)(6.29) = 85 \tag{282}$$

Standard input for the SOQ deck is 128 by 128 so this criterion is satisified.

d. SOQ input -- As a result of the above discussion, the parameters used for a bare resonator test case could be the following:

NPTS = 128

DCAL = 5.6 cm

CVM: RADC = -400 cm
DIAOUT = 4.0 cm
DIAIN = 0.0 cm

DELZ = 200.0 cm

CCM: RADC = 800.0 cm
DIAOUT = 8.0 cm
DIAIN = 0.0 cm

5. RESONATOR FOR USERS GUIDE - BARE

```
JRAHR,STMFX,P4000,T177,EC1.    RESONATORFORUSFRSGUIDEBARE
ACCOUNT(JRALT,********-***,LRO,1731)
REQUEST(TAPE8,*PF)
REQUEST(TAPE9,*PF)
GETPF(OLDPL,SOQ77128,ID=*******)
UPDATE(F,N,W,L=0)
FTN(I,LCM=I,PL=20000,L=0,A)
RETURN(OLDPL)
COPYCR,INPUT,TAPE5.
REWIND,TAPE5.
GETPF(PE8,USFRSGUIDEBARECUSM,ID=*******)
GETPF(PE9,USFRSGUIDEBARECU,ID=*******)
RFLEC(430)
LGO(PL=60000)
RFLEC(1)
PURGE(WAPE8,USERSGUIDEBARECUSM,ID=*******,LC=1)
PURGE(WAPE9,USERSGUIDEBARECU,ID=*******,LC=1)
CATALOG(TAPE8,USERSGUIDEBARECUSM,ID=*******,RP=999)
CATALOG(TAPE9,USFRSGUIDEBARECU,ID=*******,RP=999)
*EOR
*EOR
        SIMPLE CONFOCAL BARE RESONATOR - M=2, NTUBE=5.03
  $START  WWL=0.00106, NCALL=2, DCAL=6.4, NNPTS=128,
    I8=8, DDRX=0.0, DDRY=0.0, AMPGES=20.0, DGAUSS=0.0,
    RESTRT= .TRUE., PLOTS=1.0, IN=5,
    SYMTRC=.FALSE., PHIRAD=0.0, $END
        SIMPLE CONFOCAL BARE RESONATOR - M=2, NTUBE=5.03
  $CONTRL  IFLOW=2,  $END
    APPLY CVM MIRROR
  $MIROR RADC=-2000., DIAOUT=4.0, DIAIN=0.0, RMIR=.997,
    DELTA=0.0, ANGXX=0.0, ANGYY=0.0, XMPOS=0.0, YMPOS=0.0,
    DISTF=0.0,   $END
  $CONTRL IFLOW=8,  $END
       PLOT THE CVM FIELD
  $PLOT   $END
        THE CVM FIELD
  $CONTRL  IFLOW=3,  $END
```

```
  PROPAGATE THE FIELD TO THE CCM USING VAMP
 $PROPGT DELZ=1000.. WINDOX=0.1.  WINDOK=0.1. IIFG=2. IIPS=0.
  IITR=1. ROCURV=1000..  $END
 $CONTRL IFLOW=8.  $END
   PLOT THE FIELD INCIDENT ON CCM
 $PLOT  $END
   FIELD INCIDENT ON CCM
 $CONTRL  IFLOW=2.  $END
   APPLY CCM
 $MIROR RADC=4000.. DIAOUT=8..      $END
 $CONTRL IFLOW=8.  $END
   PLOT THE CCM FIELD
 $PLOT  $END
   FIELD AFTER ON CCM
 $CONTRL  IFLOW=3.  $END
 PROPAGATE THE FIELD BACK TO THE CVM USING CONSTANT AREA MESH
 $PROPGT DELZ=1000.. WINDOX=0.1.  WINDOK=0.1. IIFG=1. IIPS=0.
  IITR=0. RDCURV=0.0.  $END
 $CONTRL  IFLOW=6.  $END
   FIELD CUTOUT AND INTERPOLATION FOR THE NEXT PASS
 $CUTOUT DIREAM=4.0. OVRLAP=1.6. DXXR=0.. DYYR=0.. MAXIT=3.
   AVCUSM=0.0.  $END
 $CONTRL IFLOW=3.  $END
   PLOT THE FIELD INCIDENT ON CVM
 $PLOT  $END
   FIELD INCIDENT ON CVM
 $CONTRL  IFLOW=7.  $END
 CONVERGENCE TEST
 $CONTRL IFLOW=9.  $END
     RETURN TO MAIN PROGRAM
  $START WWL=-1..  $END
*EOR
*EOF
```

6. RESONATOR FOR USERS GUIDE - LOADED

```
JRALR.STMFX.P4000.T177.EC1.   RESONATORFORUSERSGUIDELOADED
ACCOUNT(JRALT.********-***.LRO.1731)
REQUEST(TAPE8.*PF)
REQUEST(TAPE9.*PF)
REQUEST(TAPE11.*PF)
REQUEST(TAPE12.*PF)
REQUEST(TAPE13.*PF)
GETPF(OLDPL.S0077128.ID=*******)
UPDATE(F.N.W.L=0)
FTN(I.LCM=I.PL=20000.L=0.A)
RETURN(OLDPL)
COPYCR.INPUT.TAPE5.
REWIND.TAPE5.
GETPF(TAPE8.USERSGUIDELOADEDCUSM.ID=*******)
GETPF(TAPE9.USERSGUIDELOADEDCU.ID=*******)
GETPF(TAPE11.USERSGUIDELOADEDCG11.ID=*******)
GETPF(TAPE12.USERSGUIDELOADEDCG12.ID=*******)
GETPF(TAPE13.USERSGUIDELOADEDCG13.ID=*******)
GETPF(TAPE31.OPD1131141PT8SECCONTXY.ID=*******)
RFLEC(430)
LGO(PL=60000)
RFLEC(1)
```

```
PURGE(WAPE8,USERSGUIDELOADEDCUSM,ID=********,LC=1)
PURGE(WAPE9,USERSGUIDELOADEDCU,ID=********,LC=1)
PURGE(WAPE11,USERSGUIDELOADEDCG11,ID=********,LC=1)
PURGE(WAPE12,USERSGUIDELOADEDCG12,ID=********,LC=1)
PURGE(WAPE13,USERSGUIDELOADEDCG13,ID=********,LC=1)
CATALOG(TAPE8,USERSGUIDELOADEDCUSM,ID=********,RP=999)
CATALOG(TAPE9,USERSGUIDELOADEDCU,ID=********,RP=999)
CATALOG(TAPE11,USERSGUIDELOADEDCG11,ID=********,RP=999)
CATALOG(TAPE12,USERSGUIDELOADEDCG12,ID=********,RP=999)
CATALOG(TAPE13,USERSGUIDELOADEDCG13,ID=********,RP=999)
*EOR
*EOR
      SIMPLE CONFOCAL LOADED RESONATOR - M=2, NTUBE=5.03
 $START  WWL=0.00106, NCALL=2, DCAL=6.4, NNPTS=128,
   IB=8, DDRX=0.0, DDRY=0.0, AMPGES=20.0, DGAUSS=0.0,
   RESTRT= .TRUE., PLOTS=1.0, IN=5,
   SYMTRC=.FALSE., PHIRAD=0.0, $END
      SIMPLE CONFOCAL LOADED RESONATOR - M=2, NTUBE=5.03
 $CONTRL  IFLOW=2,  $END
   APPLY CVM MIRROR
 $MIROR RADC=-2000., DIAOUT=4.0, DIAIN=0.0, RMIR=.997,
   DELTA=0.0, ANGXX=0.0, ANGYY=0.0, XMPOS=0.0, YMPOS=0.0,
   DISTF=2.E-7,   $END
 $CONTRL IFLOW=8,  $END
       PLOT THE CVM FIELD
 $PLOT   $END
      THE CVM FIELD
 $CONTRL  IFLOW=3,  $END
   PROPAGATE THE FIELD TO THE CAVITY USING VAMP
 $PROPGT DELZ=100., WINDOX=0.1,  WINDOK=0.1, IIFG=2, IIPS=0,
   IITR=0, RDCURV=1000.,  $END
 $CONTRL IFLOW=1,  $END
   APPLY GDL CAVITY
 $CAVTY1  NCAVNO=1, NSTE=4, ILR=1, NPLT=0, ZPROPI=0.,
   ZPROPO=150.,  $END
 $CAVTY2 XLEN=24.32,YLEN=11.4,  ZLEN=750.,  XMCAV=6.,   YMCAV=0.,
   NODX=190,        NODY=90,     NOSEG=3,    FLAG=11.,   MREST=0.,
   NGTYPE=0,        NGPLOT=0,    IPDEN=0,    IUSE=-1,
   T1=391.2,        T2=395.2,    T3=1284,    TN2=1333.5,
   TS=313.,         PS=.0422,    V=171380.,  PBRCH=18.,
   XN2=.8181,       XCO2=.1388, XH2O=.0146, XCO=.0044,   XO2=.0241,
   AVGAIN=.3,       $END
           USERS GUIDE LOADED RESONATOR
 $CONTRL  IFLOW=2,  $END
    APPLY CCM
 $MIROR RADC=4000., DIAOUT=8.,      $END
 $CONTRL IFLOW=8,  $END
     PLOT THE CCM FIELD
 $PLOT   $END
    FIELD AFTER CCM
 $CONTRL  IFLOW=1,  $END
  PROPAGATE THE FIELD BACK THROUGH THE CAVITY USING CONSTANT AREA MESH
 $CAVTY1  NCAVNO=1, NSTE=1, ILR=-1, NPLT=0, ZPROPI=150.,
   ZPROPO=100.,  $END
 $CONTRL  IFLOW=6,  $END
    FIELD CUTOUT AND INTERPOLATION FOR THE NEXT PASS
 $CUTOUT DIRFAM=4.0, OVRLAP=1.6, DXXR=0., DYYR=0., MAXIT=3,
    AVCUSM=-1.,  $END
```

```
 $CONTRL IFLOW=8,  $END
     PLOT THE FIELD INCIDENT ON CVM
 $PLOT   $END
     FIELD  INCIDENT ON CVM
 $CONTRL  IFLOW=7,  $END
  CONVERGENCE TEST
 $CONTRL IFLOW=9,  $END
       RETURN TO MAIN PROGRAM
 $START WWL=-1.,  $END
*EOR
*EOF
```

7. SAMPLE CODE UPDATE

The following file is included to illustrate the set of updates which would be included to add a subroutine to the existing SOQ group of subroutines. The updates are comprehensive in that they illustrate common modifications and include a namelist and subroutine within the beam quality calculation division of the SOQ code.

```
JRAZR,STMFX,P4000,T177,EC1.    ADD ZERNIKE REMOVAL TO SOQ
ACCOUNT(JRALT,********-***,LRO,1731)
GETPF(OLDPL,SOQ77128,ID=*******)
UPDATE(F,W)
FTN(I,LCM=I,PL=20000,L=0,A)
RETURN(OLDPL)
COPYCR,INPUT,TAPE5.
REWIND,TAPE5.
RFLEC(430)
LGO(PL=60000)
RFLEC(1)
*EOR
*ID ZRNIKE
*I GDL.261
      IZERN = 0
*I GDL.315
      IZERN = 0
*I SOQ77CY1.165
C            = 23  APPLY UP TO 24 ZERNIKES IN UNITS OF WAVES, READS ZERNS
*I GDL.29
      LOGICAL FRINGE
*D GDL.295,SOQ77CY1.167
C      /16 /17 /18 /19 /20 /21 /22 /23 /
     X,160,170,180,190,200,210,365,230),IFLOW
*D GDL.325,SOQ77CY1.168
C      /16 /17 /18 /19 /20 /21 /22 /23 /
     X,160,170,180,190,200,210,365,230),IFLOW
*I GDL.327
C***********************************************************
C      APPLY ZERNIKE
C***********************************************************
  230 IZERN = IZERN + 1
      IF (.NOT.INIT) GO TO 244
```

```
      FRINGE = .FALSE.
      DO 248 I=1,24
  248 P(I) = 0.
      DO 249 I=1,35
  249 PFRNG(I) = 0.
      READ (5,ZERNS)
      DO 239 I=1,35
  239 IF(PFRNG(I).NE.0.) FRINGE=.T.
      IF(.NOT.FRINGE) GO TO 241
      WRITE(6,245)
  245 FORMAT(/5X,*FRINGE COEFFICIENTS BEING CONVERTED TO SOO ORDER.*/)
      P(1) = 0.
      P(2) = PFRNG(1)
      P(3) = PFRNG(2)
      P(4) = PFRNG(3)
      P(5) = PFRNG(4)
      P(6) = PFRNG(5)
      P(7) = PFRNG(6)
      P(8) = PFRNG(7)
      P(9) = PFRNG(9)
      P(10) = PFRNG(10)
      P(11) = PFRNG(8)
      P(12) = PFRNG(11)
      P(13) = PFRNG(12)
      P(14) = PFRNG(16)
      P(15) = PFRNG(17)
      P(16) = PFRNG(13)
      P(17) = PFRNG(14)
      P(18) = PFRNG(18)
      P(19) = PFRNG(19)
      P(20) = PFRNG(25)
      P(21) = PFRNG(26)
      P(22) = PFRNG(15)
      P(23) = PFRNG(24)
      P(24) = PFRNG(35)
      IFRTST = 0
      DO 246 K=20,23
  246 IF(PFRNG(K).NE.0.) IFRTST = 1
      DO 243 K=27,34
  243 IF(PFRNG(K).NE.0.) IFRTST = 1
      IF(IFRTST.EQ.1) WRITE(6,247)
  247 FORMAT(/5X,*WARNING - FRINGE COEFFICIENTS OF ORDER 20 THROUGH 23*,
     C * AND 27 THROUGH 34 ARE IGNORED*/)
  241 DO 242 I=1,24
  242 PZSAVE(I,IZERN) = P(I)
      PZSAVE(25,IZERN) = RO
  244 CALL ZERN(PZSAVE(25,IZERN),PZSAVE(1,IZERN))
      IGNAL = 1
      GO TO 999
*D GOL.27
      DIMENSION IPLTS(50),PZSAVE(25,10),P(24),PFRNG(35)
*I GOL.33
      DATA P,PFRNG/24*0.,35*0./ , RO / 5. /
*I GOL.243
C
      NAMELIST /ZERNS/ RO,P,PFRNG
C
C     RO = RADIUS OVER WHICH ZERNIKES ARE VALID.
```

```
C      P   = ARRAY ZERNIKE COEFFICIENTS.
C      PFRNG = ARRAY FRINGE ZERNIKE COEFFICIENTS (CONVERTED TO P IN GDL).
*I LROP1.385
      SUBROUTINE ZERN(R0,P)
      LEVEL 2,CUR
      COMMON /MELT/ CUR(32768),CFIL(16512),X(128),WL,NPTS,NPY,DRX,DRY
      COMPLEX CFIL
      DIMENSION P(24)
      IF(R0.EQ.0.) GO TO 70
      DO 100 IY=1,NPY
      J1 = (IY-1)*NPTS
      YSQ = X(IY)**2
      DO 100 IX=1,NPTS
      XSQ = X(IX)**2
      INDX = IX + J1
      R = SQRT(XSQ+YSQ)
   52 THET = ATAN2(X(IY),X(IX))
      R = AMIN1(R/R0,1.)
      CT = COS(THET)
      C2T = COS(2.*THET)
      C3T = COS(3.*THET)
      C4T = COS(4.*THET)
      C5T = COS(5.*THET)

URA/R,STMFX,R400,T177,EC1.    ADD ZERNIKE REMOVAL TO SOQ
ACCOUNT(URALT,********-***,LRO,1741)
GETPF(OLDPL,SOQ77128,ID=*******)
UPDATE(F,W)
FTN(I,LCM=I,PL=20000,L=0,A)
RETURN(OLDPL)
COPYCR,INPUT,TAPE5.
REWIND,TAPE5.
RFLEC(430)
LGO(PL=60000)
RFLEC(1)
*EOR
*ID ZRNIKE
*I GDL.261
      IZERN = 0
*I GDL.315
      IZERN = 0
*I SOQ77CY1.165
C              = 23  APPLY UP TO 24 ZERNIKES IN UNITS OF WAVES. READS ZERNS
*I GDL.29
      LOGICAL FRINGE
*D GDL.295,SOQ77CY1.167
C      /16 /17 /18 /19 /20 /21 /22 /23 /
     X,160,170,180,190,200,210,365,230),IFLOW
*D GDL.325,SOQ77CY1.168
C      /16 /17 /18 /19 /20 /21 /22 /23 /
     X,160,170,180,190,200,210,365,230),IFLOW
*I GDL.327
C*****************************************************************
C      APPLY ZERNIKE
C*****************************************************************
  230 IZERN = IZERN + 1
      IF (.NOT.INIT) GO TO 244
      FRINGE = .FALSE.
      DO 248 I=1,24
  248 P(I) = 0.
```

```
      DO 249 I=1,35
  249 PFRNG(I) = 0.
      READ (5,ZERNS)
      DO 239 I=1,35
  239 IF(PFRNG(I).NE.0.) FRINGE=.T.
      IF(.NOT.FRINGE) GO TO 241
      WRITE(6,245)
  245 FORMAT(/5X,*FRINGE COEFFICIENTS BEING CONVERTED TO SOO ORDER.*/)
      P(1) = 0.
      P(2) = PFRNG(1)
      P(3) = PFRNG(2)
      P(4) = PFRNG(3)
      P(5) = PFRNG(4)
      P(6) = PFRNG(5)
      P(7) = PFRNG(6)
      P(8) = PFRNG(7)
      P(9) = PFRNG(9)
      P(10) = PFRNG(10)
      P(11) = PFRNG(8)
      P(12) = PFRNG(11)

      P(13) = PFRNG(12)
      P(14) = PFRNG(16)
      P(15) = PFRNG(17)
      P(16) = PFRNG(13)
      P(17) = PFRNG(14)
      P(18) = PFRNG(18)
      P(19) = PFRNG(19)
      P(20) = PFRNG(25)
      P(21) = PFRNG(26)
      P(22) = PFRNG(15)
      P(23) = PFRNG(24)
      P(24) = PFRNG(35)
      IFRTST = 0
      DO 246 K=20,23
  246 IF(PFRNG(K).NE.0.) IFRTST = 1
      DO 243 K=27,34
  243 IF(PFRNG(K).NE.0.) IFRTST = 1
      IF(IFRTST.EQ.1) WRITE(6,247)
  247 FORMAT(/5X,*WARNING - FRINGE COEFFICIENTS OF ORDER 20 THROUGH 23*,
     C * AND 27 THROUGH 34 ARE IGNORED*/)
  241 DO 242 I=1,24
  242 PZSAVE(I,IZERN) = P(I)
      PZSAVE(25,IZERN) = RO
  244 CALL ZERN(PZSAVE(25,IZERN),PZSAVE(1,IZERN))
      IGNAL = 1
      GO TO 999
*D GOL.27
      DIMENSION IPLTS(50),PZSAVE(25,10),P(24),PFRNG(35)
*I GOL.33
      DATA P,PFRNG/24*0.,35*0./ , RO / 5. /
*I GOL.243
C
      NAMELIST /ZERNS/ RO,P,PFRNG
C
C     RO = RADIUS OVER WHICH ZERNIKES ARE VALID.
C     P  = ARRAY ZERNIKE COEFFICIENTS.
C     PFRNG = ARRAY FRINGE ZERNIKE COEFFICIENTS (CONVERTED TO P IN GOL).
*I CROP1.385
      SUBROUTINE ZERN(RO,P)
      LEVEL 2,CUR
```

```
      COMMON /MELT/ CUR(32768),CFIL(16512),X(128),WL,NPTS,NPY,DRX,DRY
      COMPLEX CFIL
      DIMENSION P(24)
      IF(R0.EQ.0.) GO TO 60
      DO 100 IY=1,NPY
      J1 = (IY-1)*NPTS
      YSQ = X(IY)**2
      DO 100 IX=1,NPTS
      XSQ = X(IX)**2
      INDX = IX + J1
      R = SQRT(XSQ+YSQ)
   52 THET = ATAN2(X(IY),X(IX))
      R = AMIN1(R/R0,1.)
      CT = COS(THET)
      C2T = COS(2.*THET)
      C3T = COS(3.*THET)
      C4T = COS(4.*THET)
      C5T = COS(5.*THET)
      ST = SIN(THET)
      S2T = SIN(2.*THET)
      S3T = SIN(3.*THET)
      S4T = SIN(4.*THET)
      S5T = SIN(5.*THET)
      R2  = R**2
      R3 = R*R2
      R4 = R*R3
      R5 = R*R4
      R6 = R*R5
      R8 = R2*R6
      R10 = R2*R8
      DEL = P(1)   + P(2)*R*CT   + P(3)*R*ST
     A      + P(4)*(2.*R2-1.)
     B      + P(5)*R2*C2T     + P(6)*R2*S2T
     C      + P(7)*(3.*R3-2.*R)*CT   + P(8)*(3.*R3-2.*R)*ST
     D      + P(9)*R3*C3T    + P(10)*R3*S3T
     E      + P(11)*(6.*R4-6.*R2+1.)

     F      + P(12)*(4.*R4-3.*R2)*C2T   + P(13)*(4.*R4-3.*R2)*S2T
     G      + P(14)*R4*C4T     +     P(15)*R4*S4T
     H      + P(16)*(10.*R5-12.*R3+3.*R)*CT
     I      + P(17)*(10.*R5-12.*R3+3.*R)*ST
     J      + P(18)*(5.*R5-4.*R3)*C3T + P(19)*(5.*R5-4.*R3)*S3T
     K      + P(20)*R5*C5T    +    P(21)*R5*S5T
     L      + P(22)*(20.*R6-30.*R4+12.*R2-1.)
     M      + P(23)*(70.*R8-140.*R6+90.*R4-20.*R2+1.)
     N      + P(24)*(252.*R10-630.*R8+560.*R6-210.*R4+30.*R2-1.)
   60 IND2 = INDX*2
      DEL = DEL*2.*3.141592654
      COSD = COS(DEL)
      SIND = SIN(DEL)
      CURS = CUR(IND2-1)
      CUR(IND2-1) = CURS*COSD - CUR(IND2)*SIND
  100 CUR(IND2)    = CURS*SIND + CUR(IND2)*COSD
      WRITE (6,200) R0,P
  200 FORMAT (*0ZERNIKE PHASE CORRECTION APPLIED WITH NORMALIZATION*
     A * RADIUS OF *,G15.4  /* COEFFICIENTS USED P(1)-P(24)*,
     B * ARE CONSISTENT WITH THE PHASE DUE TO THE NTH TERM BEING*//
     C 20X,24H PHI(N) = 2*PI*P(N)*Z(N)//
```

```
     D * Z(N) = RF(N)*,1H*,*F(THETA)( RF(N) NORMALIZED TO 1. AT R=1.*//
     E  (1X,5G20.5))
      RETURN
   70 NOB = NPTS*NPY
      DO 80 I=1,NOB
      II=I+I
      IIM1=II-1
      CUR(IIM1) = SQRT(CUR(II)**2+CUR(IIM1)**2)
   80 CUR(II) = 0.0
      WRITE(6,300)
  300 FORMAT(//10X,*CU PHASE HAS BEEN SET TO ZERO IN SUBROUTINE ZERN*//)
      RETURN
      END
*EOR
        TEST ZERNIKE ADDITION
  $START  WWL=0.00106, NCALL=2, DCAL=15., NNPTS=128,
    IB=8, DDRX=0.0, DDRY=0.0, AMPGES=20.0, DGAUSS=0.0,
    RESTRT=.TRUE.,  PLOTS=1.0, IN=5,
    SYMTRC=.FALSE., PHIRAD=0.0, $END
      ST = SIN(THET)
      S2T = SIN(2.*THET)
      S3T = SIN(3.*THET)
      S4T = SIN(4.*THET)
      S5T = SIN(5.*THET)
      R2  = R**2
      R3 = R*R2
      R4 = R*R3
      R5 = R*R4
      R6 = R*R5
      R8 = R2*R6
      R10 = R2*R8
      DEL = P(1)   + P(2)*R*CT   + P(3)*R*ST
     A       + P(4)*(2.*R2-1.)
     B       + P(5)*R2*C2T     + P(6)*R2*S2T
     C       + P(7)*(3.*R3-2.*R)*CT   + P(8)*(3.*R3-2.*R)*ST
     D       + P(9)*R3*C3T     + P(10)*R3*S3T
     E       + P(11)*(6.*R4-6.*R2+1.)
     F       + P(12)*(4.*R4-3.*R2)*C2T   + P(13)*(4.*R4-3.*R2)*S2T
     G       + P(14)*R4*C4T      +    P(15)*R4*S4T
     H       + P(16)*(10.*R5-12.*R3+3.*R)*CT
     I       + P(17)*(10.*R5-12.*R3+3.*R)*ST
     J       + P(18)*(5.*R5-4.*R3)*C3T + P(19)*(5.*R5-4.*R3)*S3T
     K       + P(20)*R5*C5T    +    P(21)*R5*S5T
     L       + P(22)*(20.*R6-30.*R4+12.*R2-1.)
     M       + P(23)*(70.*R8-140.*R6+90.*R4-20.*R2+1.)
     N       + P(24)*(252.*R10-630.*R8+560.*R6-210.*R4+30.*R2-1.)
   60 IND2 = INDX*2
      DEL = DEL*2.*3.141592654
      COSD = COS(DEL)
      SIND = SIN(DEL)
      CURS = CUR(IND2-1)
      CUR(IND2-1) = CURS*COSD - CUR(IND2)*SIND
  100 CUR(IND2)    = CURS*SIND + CUR(IND2)*COSD
      WRITE (6,200) R0,P
  200 FORMAT (*0ZERNIKE PHASE CORRECTION APPLIED WITH NORMALIZATION*
     A * RADIUS OF *,G15.4  /* COEFFICIENTS USED P(1)-P(24)*,
     B * ARE CONSISTENT WITH THE PHASE DUE TO THE NTH TERM BEING*//
     C 20X,24H PHI(N) = 2*PI*P(N)*Z(N)//
     D * Z(N) = RF(N)*,1H*,*F(THETA)( RF(N) NORMALIZED TO 1. AT R=1.*//
     E  (1X,5G20.5))
```

```
      RETURN
   70 NOR = NPTS*NPY
      DO 80 I=1,NOR
      II=I+I
      IIM1=II-1
      CUR(IIM1) = SQRT(CUR(II)**2+CUR(IIM1)**2)
   90 CUR(II) = 0.0
      WRITE(6,300)
  300 FORMAT(//10X,*CU PHASE HAS BEEN SET TO ZERO IN SUBROUTINE ZERN*//)
      RETURN
      END
*EOR
        TEST ZERNIKE ADDITION
  $START  WWL=0.00106, NCALL=2, DCAL=15., NNPTS=128,
    IB=8, DDRX=0.0, DDRY=0.0, AMPGES=20.0, DGAUSS=0.0,
    RESTRT=.TRUE., PLOTS=1.0, IN=5,
    SYMTRC=.FALSE., PHIRAD=0.0, $END
        TEST ZERNIKE ADDITION
 $CONTRL IFLOW=4,  $END
      APERTURE THE PLANE WAVE TO 10. CM.
 $APTUR  DOUT=10.,  DIN=0.,  $END
 $CONTRL IFLOW=8,  $END
      PLOT THE INITIAL PLANE WAVE
 $PLOT  $END
        INITIAL PLANE WAVE
 $CONTRL  IFLOW=23,  $END
      APPLY SPECIED ZERNIKES
 $ZERNS R0=5, P(4)=.1, P(5)=.1, P(6)=.1,  $END
 $CONTRL IFLOW=8,  $END
      PLOT THE ZERNIKED PLANE WAVE
 $PLOT  $END
  ZERNIKED PLANE WAVE
 $CONTRL  IFLOW=23,  $END
     REMOVE SPECIED ZERNIKES
 $ZERNS R0=5, PERNG(3)=-.1, PERNG(4)=-.1, PERNG(5)=-.1,  $END
 $CONTRL IFLOW=8,  $END
      PLOT THE DEZERNIKED PLANE WAVE
 $PLOT  $END
  DEZERNIKED PLANE WAVE
 $CONTRL  IFLOW=9,  $END
     RETURN TO MAIN
 $START  WWL=-1.,  $END
*EOR
```

To obtain source printouts of the SOQ code, the user must run the CDC update program. The compile file may be used as a source listing or if the user so desires he may run the Fortran compiler on the code to obtain a compiled version or listing along with any desired Fortran compiler options supported under the CDC NOS/BE system. The file output will contain the desired listings. The following job setup is include as a guide:

Job Card

Account Card

Attach, OLDPL, SOQ77128, ID=

Update, F.

FTN.

REFERENCES

1. Sziklas, E.A., et al., System Optical Quality Study Phase I - Problem Definition, AFWL-TR-73-231, Pratt & Whitney Aircraft, June 1974.

2. Sziklas, E.A., and Siegman, A.E., "Mode Calculations in Unstable Resonators with Flowing Saturable Gain. 2: Fast Fourier Transform Method," Applied Optics, Vol. 14, pp. 1874-1899, August 1975.

3. Siegman, A.E. and Sziklas, E.A., "Mode Calculations in Unstable Resonators with Flowing Saturable Gain. 1: Hermite-Gaussian Expansion," Applied Optics, Vol. 13, p. 2775, 1974.

4. Krupke, W.F. and Sooy, W.R., "Properties of an Unstable Confocal Resonator CO_2 Laser System," IEEE Journal of Quantum Electronics, Vol. QE-5, No. 12, p. 579, December 1969.

5. Kogelnik, H. and Li, T., "Laser Beams and Resonators," Proc. IEEE, Vol. 54, pp. 1312-1329, October 1966.

6. Siegman, A.E., "Stabilizing Output with Unstable Resonators," Laser Focus, May 1961.

7. Goodman, J.W., Introduction to Fourier Optics, McGraw-Hill, New York, 1968.

8. Sziklas, E.A. and Siegman, A.E., "Diffraction Calculations Using Fast Fourier Methods," Proc. IEEE, Vol. 62. pp. 410-412, March 1974.

9. Higgins, R.J., "Fast Fourier Transform; An Introduction with Some Minicomputer Experiments," AJP, 44, 1976.

10. Biblarz, O. and Fuhs, A.E., "Laser Cavity Density Changes with Kinetics of Energy Release," AIAA Journal, Vol. 12, p. 1083, August 1974.

11. Fuhs, A.E., "Quaside Area Rule for Heat Addition in Transonic and Supersonic Flight Regimes," AFAPL-TR-72-10, Air Force Aero Propulsion Laboratory, WPAFB, Ohio, 1972.

12. Tsien, H.E. and Milton Beilock, "Heat Source in a Uniform Flow," Journal of the Aeronautical Sciences, p. 756, December 1949.

13. Siegman, A.E. and Miller, H.Y., "Unstable Optical Resonator Loss Calculations Using the Prony Method," Applied Optics, Vol. 9, p. 2729, 1970.

14. Siegman, A.E., An Introduction to Lasers and Masers, McGraw-Hill, New York, 1971.

15. Humphreys, W.W. and Wick, R.V., "Change in Optical Path Length Near a Hot Mirror Surface," Laser Digest, AFWL-TR-75-140, 1975, p. 9.

16. Bergland, G.D., "A Guided Tour of the Fast Fourier Transform," IEEE Spectrum 6, 1969.

17. Cooley, J.W., Lewis, P.A.W., Welch, P.D., "The Finite Fourier Transform," IEEE Trans. Audio Electroacoust, AU-17, 1969.

DATE

FILMED